

STUDIES IN LOGIC
AND
THE FOUNDATIONS OF MATHEMATICS

VOLUME 139

S. ABRAMSKY / S. ARTEMOV / R.A. SHORE / A.S. TROELSTRA
EDITORS

*Handbook of
Recursive Mathematics*

*Volume 2, Recursive Algebra,
Analysis and Combinatorics*

Edited by

Yu. L. ERSHOV
S.S. GONCHAROV
A. NERODE
J.B. REMMEL

Associate Editor

V.W. MAREK

ELSEVIER

AMSTERDAM • LAUSANNE • NEW YORK • OXFORD • SHANNON • SINGAPORE • TOKYO

HANDBOOK OF
RECURSIVE MATHEMATICS

Volume 2:
Recursive Algebra, Analysis and Combinatorics

STUDIES IN LOGIC
AND
THE FOUNDATIONS OF MATHEMATICS
VOLUME 139

Honorary Editor:

P. SUPPES

Editors:

S. ABRAMSKY, *London*
S. ARTEMOV, *Moscow*
R.A. SHORE, *Ithaca*
A.S. TROELSTRA, *Amsterdam*



ELSEVIER
AMSTERDAM • LAUSANNE • NEW YORK • OXFORD • SHANNON • SINGAPORE • TOKYO

HANDBOOK OF RECURSIVE MATHEMATICS

Volume 2: Recursive Algebra, Analysis and Combinatorics

Edited by

Yu. L. ERSHOV

*Sobolev Institute of Mathematics
pr. Akademika Koptuga, 4
Novosibirsk, 630090, Russia*

A. NERODE

*Department of Mathematics
Cornell University
Ithaca, NY 14853-7901, U.S.A.*

S.S. GONCHAROV

*Sobolev Institute of Mathematics
pr. Akademika Koptuga, 4
Novosibirsk, 630090, Russia*

J.B. REMMEL

*Department of Mathematics
University of California at San Diego
La Jolla, CA 92093, U.S.A.*

Associate Editor

V.W. MAREK

*Department of Computer Science
University of Kentucky
Lexington, KY 40506, U.S.A.*



1998

ELSEVIER

AMSTERDAM • LAUSANNE • NEW YORK • OXFORD • SHANNON • SINGAPORE • TOKYO

ELSEVIER SCIENCE B.V.
Sara Burgerhartstraat 25
P.O. Box 211, 1000 AE Amsterdam, The Netherlands

Library of Congress Cataloging in Publication Data
A catalog record from the Library of Congress has been applied for.

ISBN: 0 444 50106 1 volume 2 (SiL vol. 139)

ISBN: 0 444 50003 0 volume 1 (SiL vol. 138)

ISBN: 0 444 50107 X (set: vols. 1 & 2)

© 1998 Elsevier Science B.V. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written permission of the publisher, Elsevier Science B.V., Copyright & Permissions Department, P.O. Box 521, 1000 AM Amsterdam, The Netherlands.

Special regulations for readers in the U.S.A. – This publication has been registered with the Copyright Clearance Center Inc. (CCC), 222 Rosewood Drive, Danvers, MA 01923. Information can be obtained from the CCC about conditions under which photocopies of parts of this publication may be made in the U.S.A. All other copyright questions, including photocopying outside of the U.S.A., should be referred to the copyright owner, Elsevier Science B.V., unless otherwise specified.

No responsibility is assumed by the publisher for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions or ideas contained in the material herein.

☺The paper used in this publication meets the requirements of ANSI/NISO Z39.48-1992
(Permanence of Paper).

Printed in The Netherlands.

Contents

Introduction to the Handbook of Recursive Mathematics	vii
<i>Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel</i>	

Part One Recursive Model Theory

1	Pure Computable Model Theory	3
	<i>V. S. Harizanov</i>	
2	Elementary Theories and their Constructive Models	115
	<i>Yu. L. Ershov and S. S. Goncharov</i>	
3	Isomorphic Recursive Structures	167
	<i>C. J. Ash</i>	
4	Computable Classes of Constructive Models	183
	<i>V. P. Dobritsa</i>	
5	Σ -Definability of Algebraic Structures	235
	<i>Yu. L. Ershov</i>	
6	Autostable Models and Algorithmic Dimensions	261
	<i>S. S. Goncharov</i>	
7	Degrees of Models	289
	<i>J. F. Knight</i>	
8	Groups of Computable Automorphisms	311
	<i>A. S. Morozov</i>	
9	Constructive Models of Finitely Axiomatizable Theories	347
	<i>M. G. Peretyat'kin</i>	
10	Complexity Theoretic Model Theory and Algebra	381
	<i>D. Cenzer and J. B. Remmel</i>	
11	A Bibliography of Recursive Algebra and Recursive Model Theory	515
	<i>I. Kalantari</i>	
12	A Bibliography of Recursive Analysis and Recursive Topology	583
	<i>V. Brattka and I. Kalantari</i>	

Part Two Recursive Algebra, Analysis, and Combinatorics

13	Π_1^0 Classes in Mathematics <i>D. Cenzer and J. B. Remmel</i>	623
14	Computability Theory and Linear Orderings <i>R. G. Downey</i>	823
15	Computable Algebras and Closure Systems: Coding Properties <i>R. G. Downey and J. B. Remmel</i>	977
16	A Survey of Recursive Combinatorics <i>W. Gasarch</i>	1041
17	Constructive Abelian Groups <i>N. G. Khisamiev</i>	1177
18	Recursive and On-Line Graph Coloring <i>H. A. Kierstead</i>	1233
19	Polynomial-Time Computability in Analysis <i>K. Ko</i>	1271
20	Generally Constructive Boolean Algebras <i>S. P. Odintsov</i>	1319
21	Reverse Algebra <i>S. G. Simpson and J. Rao</i>	1355

Introduction to the Handbook of Recursive Mathematics

Y. L. Ershov, S. S. Goncharov,
A. Nerode, and J. B. Remmel

Preface

Recursive or Computable Mathematics is the study of the effective or computable content of the techniques and theorems of Mathematics. Recursive Mathematics has been an active area of research for the last 25 years. Its tools are the techniques of modern Computability Theory. These tools have been applied to analyze the effective content of results in a wide variety of mathematical fields including Algebra, Analysis, Topology, Combinatorics, Logic, Model Theory, Algebraic Topology, and Mathematical Physics.

This volume consists of surveys and new results from many of the leading researchers in the field. Almost all of the major subfields of Recursive Mathematics are represented in the volume. There are contributions which analyze the effective content of results from Algebra, Analysis, Combinatorics, Orderings, Logic, Non-monotonic Logics, Topology and Model Theory.

In this introduction, we will describe the historical roots of Recursive Mathematics as well as summarize some of the major themes that have arisen in the study of Recursive Mathematics and that are covered by the articles in this volume. We also provide a brief description of the articles of this volume. Finally we will end with what the editors believe are promising areas for further research.

Historical roots of recursive and constructive mathematics

Mathematics consists of constructions of new mathematical objects from old and of proofs that these objects have certain properties and lack other properties. Cantor's work on set theory in the late nineteenth century presented to the mathematical world what Hilbert called the paradise of set theory. Although Cantor's work was controversial at that time, the set theoretic point of view is at present widely accepted. Thus in the twentieth century, most working mathematicians have accepted Zermelo-Fraenkel set theory as the foundation of mathematics within which their work takes place. That is, almost all working mathematicians generally accept that Zermelo-Fraenkel set theory provides a foundation for mathematics with the caveat that occasionally one might have to extend ZF or ZFC by new axioms to accommodate certain proofs and constructions.

Within Zermelo-Fraenkel set theory, sets are built up by operations which construct sets from sets such as union, intersection, and power set. In addition, one employs comprehension-like schemes which assert that a set exists consisting of all elements having a property expressed in the language of set theory. The expression of the property in the language of set theory may itself involve sets as is the case with the axiom of replacement. Moreover, one assumes that the basic objects, sets, are extensional. That is, if two different properties are satisfied by the same elements, and one property defines a set, then the other property defines the same set.

However there are much narrower conceptions of mathematical objects, constructions, and proofs proposed by finitists and constructivists. In one way or another, they can be more or less adequately described within the classical Zermelo-Fraenkel set theory, but some of them are better viewed as alternate foundations of mathematics.

The earliest instance of such constructivist tendencies is the classical Greek fascination with what can be constructed by limited means. For instance consider the concept of a ruler and compass construction. Here one is naturally led to ask questions like: "What are the ruler and compass constructible regular polygons?" or "Is a segment the length of a circumference of circle ruler and compass constructible from a segment the length of a diameter?".

Algebra as developed by the Greek Diophantus and by the Persian al-Khwârisimî and his successors was wholeheartedly constructive. They wanted to find formulas for the solution of algebraic equations. In fact, it is thought by some that the word “algorithm” comes from the name al-Khwârisimî. Descartes’ seventeenth century coordinatization of Geometry was intended as a reduction of geometric construction problems to the problem of solving algebraic equations. This endeavor was highly constructive. In fact, the explicit formula-based approach to Algebra continued into the nineteenth century to the time of Galois and Abel. By then negative results could be obtained since, through Lagrange, the notion of a solution by radicals of the fifth degree equation became precise and Galois’ theory showed there were no such solutions for the general fifth degree equation.

Of the two strongest algebraists of the middle of the nineteenth century, Kronecker and Dedekind, Kronecker was the one to develop explicit formula and algorithms for all of his complex and Diophantine Algebraic Geometry. Kronecker’s philosophy was that the true subject matter of mathematics is concretely representable structures whose operations are combinatorial. Dedekind preferred a set theoretic approach, which was not algorithmic. One need merely contrast Dedekind’s set-theoretic definition of prime ideals and primary ideal factorizations with Kronecker’s wholly constructive description via polynomials found by factoring algorithms. Dedekind explicitly said that he wanted set-theoretic definitions which are independent of the algorithm used. Thus Dedekind distinguishes himself from Kronecker and started to lead us away from algorithmic mathematics and toward Cantor’s absolutist set theory.

In Analysis, Newton and Leibniz put great emphasis on solving geometric and physical problems by translating them to purely symbolic forms. Newton especially emphasized power series. For physicists ever since, publishing proofs without algorithms is just not satisfactory. Almost all of Analysis was highly constructive till the Weierstrassian revolution in the 1850’s when exact definitions and proofs were developed for real n -dimensional Analysis and pure existence proofs using compactness arguments started to become popular. After the time of Dedekind in Algebra and Weierstrass in Analysis, general non-algorithmic set-theoretic methods took hold.

However, very early on, mathematicians, including Cantor himself, realized that there were paradoxes within naive set theory. The response to these paradoxes led to an extensive interest in the foundation of mathematics by many of the leading mathematicians including the likes of Hilbert

and Poincaré. Of course, one reaction to the paradoxes was to develop a set of axioms for set theory which was free of the paradoxes and yet was strong enough to capture almost all mathematical constructions. This work led to the axiomatization of set theory by Zermelo and Fraenkel, and by Gödel and Bernays. Quite another reaction was to abandon naive set theory and return a completely constructive foundation for mathematics as reflected in the work of Brouwer and Heyting. In 1918, Brouwer [11, 12] announced his constructivism, and played a role in Analysis similar to that of Kronecker in Algebra. He developed n -dimensional calculus with proofs and constructions that can actually be carried out by (mental) algorithms. Since he denounced classical logic and the resulting classical mathematics as being wrong-headed, rather than simply saying that it was a less constructive species of thought, he encountered a great deal of hostility. One of the reasons for this hostility is that he used all of the words of the conventional mathematician, both for mathematics and for proofs, but with a different meaning, and asked everyone else to accept his meanings. He was understandably hostile to formalizing constructive logic, because as a humble human being, he thought that the human mind might extend his or anyone else's notion of construction by finding new ones at any time.

Hilbert's Program and the development of computable functions

The conflict between the philosophy of constructive mathematics which was being pursued by Brouwer [11, 12] and by his student Heyting [65, 66] and the overwhelming success of abstract methods, in part developed by Hilbert himself, lead, in the 1920's, to Hilbert's program to provide a finitistic proof of the consistency of mathematics.

Of course, Hilbert's program of proving mathematics consistent by finitary means was generally accepted as coming to an end with Gödel's famous incompleteness results [51]. Indeed, Gödel observed that his theorems applied to any "sufficiently rich" theory and that the notion of "sufficiently rich" could be rephrased as systems which could compute a wide class of functions by employing the proof procedures of the system. Within a few years, the idea of a computable function on the integers had emerged through the efforts of Church [19], Kleene [79], and Turing [154]. Later significant versions

of the definition of computability were published by Post [120], Markov [97] and Mal'tsev [92]. By the end of the 1940's, all the proposed definitions up to that time had been proven equivalent and the formal definition of computable function was widely accepted as a plausible formalization of the intuitive notion of a function of integers whose values can be computed by a fixed program or algorithm. Moreover great effort was spent to show that every definition of program that one could think of yielded a function which was computable in the formal sense. Indeed, it remains the case today that no one has produced a function on the natural numbers which can be computed by an algorithm or program that is not included within the formal definition of computable function. However, just as Brouwer was humble enough to understand that there may be people with new constructions which we will all accept, so too we have to admit that it is possible that some computation scheme, not hitherto thought of, might extend the class of computable functions. But till then, we accept Church's thesis that the formal definition of recursive function captures the intuitive notion of computable function.

Accepting Church's Thesis has profound implications as well as limitations. To understand these implications and limitations, consider Hilbert's 10-th problem which was to find a procedure which, in a finite number of steps, determines whether or not a Diophantine equation has an integral solution. We regard Matijasevič [100] as having solved this problem because we have accepted that such a procedure must be represented by a program computing a computable function with the currently accepted definition. If someone in the future extends the notion of computable function beyond our current formal definition of computable or recursive function, we would have to revisit Hilbert's 10-th problem. The proof of the unsolvability of the word problem for groups, Church's theorem on the undecidability of formalized arithmetic itself, etc., all have this limitation, as does most of the work presented in this volume.

We should note that Hilbert never gave up on his program. Gödel said that perhaps the only way to close the issue is to find universally acceptable axioms for finitary methods and then to show that consistency cannot be proved by any method obeying the axioms. This has not been done yet. Heyting's formalization of constructive proofs by intuitionistic predicate calculus was not accepted as the last word by Brouwer for the same reasons as expressed above. That is, one cannot rule out the possibility that new constructive rules of reasoning might be discovered tomorrow and accepted universally. Nevertheless, Heyting's formalization remains important since it

has been seen to have very deep connections with Church's lambda calculi, and the Church-Curry theories, in their typed reincarnations, have had wide impact in Computer Science. In the end, the significance of both the formal definitions of computable functions and of intuitionistic logic will have to rest on the insights they give us about the nature of constructions and their usefulness in mathematics, computer science, linguistics, and other areas of science, and not on their problematical philosophical significance as final descriptions of intuitive computability and intuitive constructive reasoning.

While the historical roots and motivation of Recursive Mathematics can be found in constructivist philosophies of mathematics, Recursive Mathematics does not lie within Constructive Mathematics. Researchers in Recursive Mathematics use classical reasoning unacceptable to constructivists to prove theorems about the non-existence of computable procedures. They accept Church's Thesis that the modern definition of a recursive function captures all functions which are intuitively computable. Constructivists simply do not accept the classical logic behind Recursive Mathematics. Bishop was the first to capture the constructive content of modern Functional Analysis, (Brouwer's work was finite dimensional). Bishop did not view Recursive Mathematics as within the constructive philosophy of mathematics espoused in his book on Constructive Analysis [6]. He was, however, interested in Recursive Analysis and was helpful to researchers in Recursive Mathematics, including Metakides, Nerode, and Remmel. Despite the fact that Recursive Mathematics is not part of Constructive Mathematics, the work on Recursive Mathematics does have interesting implications for Constructive Mathematics. For example, many of the counterexamples in the literature of Recursive Mathematics can easily be translated into counterexamples accepted by a constructivist as espoused say in the work of Troelstra and van Dalen [152, 153]. The converse is also true that constructivist counterexamples can be used as the basis for counterexamples in Recursive Mathematics.

Development of recursive mathematics

With the development of a coherent notion of computable function, it was quite natural that researchers would begin to apply it to other areas of mathematics.

In the 1930's, Church and Kleene raised the question of effective content of the theory of ordinals, namely, which ordinals and operations on ordinals

are constructive from the point of view of recursive functions? This they called the theory of constructive ordinals. The Church-Kleene constructive ordinals were later shown by Markwald [99] to be the ordinals represented by recursive well-orderings. What is noteworthy is the non-extensional character of the Church-Kleene theory. Their theory is a theory of recursive operations on notations (indices) for ordinals. The questions of whether a number is a notation for an ordinal, or whether two notations for ordinals are notations for the same ordinal, are highly non-constructive. This means that all algorithms should act on notations (indices) for ordinals, not on the ordinals themselves. Thus Church-Kleene theory of recursive ordinals contains a basic insight which is shared with constructivists, namely, constructive operations should not be restrained to be extensional.

Brouwer-Heyting intuitionistic mathematics continued on in the 1930's. Brouwer influenced the algebraist van der Waerden. In his 1937 book, *Modern Algebra* [157], supposedly based on lectures of the highly abstract algebraist Emil Artin, he said in a most un-Artinian manner "*a field Δ is given explicitly* if its elements are uniquely represented by distinguishable symbols with which addition, subtraction, multiplication, and divisions can be performed by a finite number of operations", and proved that if a field Δ is given explicitly, then every simple algebraic extension $\Delta(t)$ and every simple algebraic extension $\Delta(\nu)$ defined by an irreducible polynomial φ is given explicitly. On the other hand, van der Waerden states that the construction of a splitting field requires that the field have a factorization algorithm for polynomials. He states explicitly that there is no known universal method for factorization and "there are reasons for the assumption that such a general method is impossible" and refers to his 1930 paper [155] as justification. With the development of recursion theory it was natural to apply its definitions and methods, and analogous results were proven in the recursive mathematics context by Frölich and Shepherdson [48] in the 1950's.

As for determining the effective content of Analysis, in the 1940's Goodstein [58] asked what it means to be a primitive recursively computable function of reals. Pre-World War II work of Banach-Mazur [101] was published in the same area in 1963.

The development of Recursion Theory in the 1940's and early 50's by Kleene [80, 82], Post [121], Peter [119], Myhill [109], Rice [132], and many others, provided the basic tools for the first results in Recursive Mathematics which appeared in the 1950's and early 60's. In particular, Frölich and Shepherdson [48] and Rabin [124, 125] provided explicit counterexamples of

recursive fields which failed to have factorization algorithms, which justified van der Waerden's claim. Spector analyzed the effective content of recursive well orderings [144]. Kreisel analyzed the Cantor-Bendixson theorem [85]. Dekker and Myhill began to develop Isol Theory which can be viewed as an effective version of the theory of cardinals [33]. Lacombe [88], Goodstein [58] and Markov [98] started their work on Recursive Analysis. Similarly Julia Robinson [133] and Davis, Putnam and Robinson [29] started their work on Hilbert's 10-th problem which was finally culminated in Matijasevič's result [100, 28] that there is no recursive algorithm to decide whether an arbitrary Diophantine equation has an integer solution.

Modern history of Recursive Mathematics

While there was not a formally recognized subject of Recursive Mathematics, a number of results in Recursive Mathematics appeared in 1960's and early 70's. For example, in Isol Theory, the work of Ellentuck [34] and Nerode [110] answered many of the fundamental questions about the theory of effective cardinality on the natural numbers so that researchers begin to apply effective cardinality theory to algebraic structures. In particular, Crossley [23], and independently Manaster, developed a theory of constructive order types. Ellentuck [36], Applebaum [2] and Hasset [62] developed a theory of isolic groups, Dekker [30, 31] developed isolic vector spaces. Finally Crossley and Nerode gave a general theory for isolic structure in their book Combinatorial Functors [25]. Remmel [126] extended Hay's theory of co-r.e. isolos to the general setting of Crossley and Nerode. Similarly there was considerable work on the effective content of theorems of combinatorics. Jockusch [72] studied the effective content of Ramsey's theorem. Jockusch and Soare produced their fundamental papers on Π_1^0 -classes [74, 73] which can be viewed as the study of the effective content of König's lemma on trees. Manaster and Rosenstein studied the effective content [95, 96] of Hall's matching theorem and graph colorings. The study of the effective content of Analysis continued to flourish throughout this period. In particular, there was the work of Šanin [137] and others in his Leningrad school, including Orevkov [118] and Ceitin [14]. Other contributions include the work of Kušner [87] and others in Moscow who were influenced by Markov. In the West, key contributions were made by Aberth [1], Hauck [63], and, much later in the 1980's, by Pour-El and Richards [122].

The foundations for much of the work represented in this volume can be traced to fundamental work by Ershov and Nerode in the early 1970's. First in Russia, Ershov building on Mal'tsev's work on constructive algebra [92] published his theory of enumerations [38, 39] in which he defined a *constructivization* of a model $\mathcal{A}$ with universe A to be a surjective function $\alpha : \omega \rightarrow A$ such that for any atomic formulae φ in the underlying language $\mathcal{L}$, the relations $\{\bar{a} : \mathcal{A} \models \varphi(\alpha(\bar{a}))\}$ are recursive uniformly in φ . In fact, Ershov's theory of constructive models was first written in 1967 at Novosibirsk but was not published until after 1973. Ershov's work led to a vast amount of research on constructive Model Theory and Algebra in the former Soviet Union. This effort was lead by Ershov, Goncharov and their students Peretyat'kin, Nurtazin, Dobritsa, Khisamiev, Kudaibergenov, Ventsov, Fedoryaev, Morozov, and many others.

Independently, at Cornell University in the United States in 1972, Nerode began his program to determine the effective content of mathematical constructions and started to develop a systematic theory of recursive structures. Let $\varphi_{e,n}$ denote the partial recursive function of n variables computed by the e -th Turing machine. Here we say that a structure

$$\mathcal{A} = (A, \{R_i^{\mathcal{A}}\}_{i \in S}, \{f_i^{\mathcal{A}}\}_{i \in T}, \{c_i^{\mathcal{A}}\}_{i \in U}),$$

(where the universe A of $\mathcal{A}$ is a subset of the natural numbers ω), is *recursive* if A is a recursive subset of ω , S , T , and U are initial segments of ω , the set of relations $\{R_i^{\mathcal{A}}\}_{i \in S}$ is uniformly recursive in the sense that there is a recursive function G such that for all $i \in S$, $G(i) = [n_i, e_i]$ where $R_i^{\mathcal{A}}$ is an n_i -ary relation and φ_{e_i, n_i} computes the characteristic function of $R_i^{\mathcal{A}}$, the set of functions $\{f_i^{\mathcal{A}}\}_{i \in T}$ is uniformly recursive in the sense that there is a recursive function F such that for all $i \in T$, $F(i) = [n_i, e_i]$ where $f_i^{\mathcal{A}}$ is an n_i -ary function and φ_{e_i, n_i} restricted to A^{n_i} computes $f_i^{\mathcal{A}}$, and there is a recursive function interpreting the constant symbols in the sense that there is a recursive function H such that for all $i \in U$, $H(i) = c_i^{\mathcal{A}}$. Note that if $\mathcal{A}$ is a recursive structure, then the atomic diagram of $\mathcal{A}$ is recursive. Thus a recursive structure which is isomorphic to a structure $\mathcal{B}$ can be viewed a constructivization of $\mathcal{B}$. It follows that recursive structures and constructivizations are essentially interchangeable.

In particular, in a series of papers with his student Metakides [102, 103, 104, 105], Nerode introduced the systematic use of the finite injury priority

argument applied to algebraic requirements to construct recursive models which could be used to give effective counterexamples to classical mathematical theorems. This allowed the whole technology of modern Recursion Theory as developed by Friedberg, Rogers, Sacks, Lachlan, Soare, Jockusch, Lerman, and others, to be applied to these problems. Many applications of the priority argument to Recursive Mathematics were produced by Nerode's students at that time: Remmel, Metakides, Millar, Kalantari, Retzlaff, Lin, and subsequently by many others in North America, and also by Crossley, Ash, Downey, Moses, Hird and others in Australia. Another theme that emerged from this work was a systematic theory of the lattice of recursively enumerable substructures of a recursive structure (see the 1982 survey paper by Nerode and Remmel [112]).

During the late 1970's and 80's, Cold War politics allowed almost no communications between the schools in Novosibirsk of Ershov and the school of Nerode in the West, so that often results in Recursive Model Theory and Algebra were duplicated. Nevertheless, 25 years of work by researchers associated in one way or another with these two schools have generated a large body of results which is the main subject of most of the papers in this volume. Unfortunately, it is impossible within a single volume to cover all the areas of Recursive Mathematics and its connections with other subjects such as Reverse Mathematics, Constructive Set Theory, and Decidability Theory. However, we feel that the articles in this volume will provide the reader with a good overview of a significant portion of the work that has been produced and continues to be produced in Recursive Mathematics. In addition, the bibliographies on Recursive Algebra and Model Theory compiled by Kalantari and on Recursive Analysis compiled by Brattka and Kalantari are valuable resources for current and future researchers in Recursive Mathematics. Recursive Mathematics continues to be a vital field and our hope is that this volume will inspire others to deepen our understanding and broaden the fields of application of Recursive Mathematics.

Some general themes of Recursive Mathematics

A number of important common mathematical themes emerged from the past work on Recursive Mathematics which the reader will find reflected in this volume. These include the following.

1. Fix a given recursive structure, ideally one which is universal for a large class of recursive structures, and study the complexity of various model-theoretical and algebraic constructions on that structure.
2. Find necessary and sufficient conditions for the existence of recursive or constructive models of a theory with given properties.
3. Classify the class of recursive models within a classical isomorphism type that are unique up to recursive isomorphisms. More generally, find necessary and sufficient conditions for existence of recursive or constructive representations of a given structure with nonequivalent algorithmic properties.
4. Study the lattice of recursively enumerable substructures of a given recursive structure. In particular, investigate the similarities and differences with the lattice of r.e. substructures of an algebraic structure and the so-called Post's zoo of maximal, simple, h-simple, hh-simple sets in the lattice of r.e. sets.
5. Determine whether the class of all recursive models of a given structure is computable.
6. Study the complexity of the set of solutions of a recursive instance of a classical combinatorial, algebraic or analytic problem.
7. Study the possible degrees of models of a given theory or the set of degrees of all structures which are isomorphic to a given structure.
8. Study the differences between r.e. presented structures and recursive structures. Here, an r.e. presented structure consists of a recursive structure modulo an r.e. congruence relation. For example, an r.e. Boolean algebra can be represented as a recursive Boolean algebra modulo an r.e. ideal. More generally, study the differences between recursive structures and structures recursive in some oracle set A .

In the first area, many algebraic structures have been studied. In the West, the computability of ordered sets was studied by Ash, Case, Chen, Crossley, Downey, Feiner, Feldman, Fellner, Hay, Hingston, Hird, Jockusch, Kierstead, Knight, Lerman, Manaster, Metakides, McNulty, Moses, Remmel, Richter, Rosenstein, Roy, Schmerl, Schwarz, Soare, Tennenbaum, Trotter

and Watnick; the computability of vector spaces by Ash, Dekker, Downey, Guhl, Guichard, Hamilton, Kalantari, Remmel, Retzlaff, Shore, Smith and Welch; the computability of rings and fields by Ash, Hodges, Jockusch, MacIntyre, Madison, Marker, Metakides, Nerode, Mines, Remmel, Rosenthal, Seidenberg, Shlapentokh, Smith, Staples, Tucker and van den Dries; the computability of the structures with a dependence relation by Baldwin, Downey, Metakides, Nerode, and Remmel. Other mathematical structures that were also extensively studied include groups by Ash, Barker, Ge, Kent, Knight, Lin, Oates, Richards, Richman and Smith; graphs by Aharoni, Bean, Beigel, Burr, Carstens, Gasarch, Golze, Kierstead, Lockwood, Manaster, Magidor, P\"appinghaus, Remmel, Rosenstein, Schmerl and Shore; Boolean algebras by Carroll, Downey, Feiner, La Roche, Remmel, Soare and Thurber; topological spaces by Kalantari, Leggett, Remmel, Retzlaff and Weitkamp. Computable Ramsey's theory has been studied by Clote, Hummel, Jockusch, Seetapun, Simpson, Solovay and Specker. In the East, Goncharov and his students extensively studied recursive Boolean algebras and orderings, Morozov studied groups of recursive automorphisms, Khisamiev studied Abelian p -groups establishing connections between constructivizability and Ulm's quotients, and Odintsov studied semi-lattices of recursively enumerable subalgebras.

For the second type of problem, the following principal results were obtained: the Ershov theorems about kernels and the theorem about the existence of constructive models for a theory with finite obstacles, Baur's theorem about the existence of constructive model for $\forall\exists$ -theories, the Goncharov and Harrington theorem about the decidability of prime models, the Morley theorem about the decidability of homogeneous models, the Goncharov-Peretyat'kin criterion for decidability of homogeneous models, the Khisamiev-Millar theorems about omitting recursive types, and theorems due to Lachlan, Peretyat'kin, Millar, and Ash-Read about decidability of Ehrenfeucht theories.

For the third problem various equivalence types on constructivizations of models were studied. In particular: recursive equivalence, autoequivalence, algebraic equivalence, program equivalence, uniform equivalence were investigated by Goncharov, Peretyat'kin, Nurtazin, Ash, Nerode, Ventsov, Uspenskiĭ, Remmel, Kudinov and others. In this direction, powerful criteria for the existence of nonautoequivalent recursive representations of models and criteria for autostability were found by Goncharov, Nurtazin, and Kudinov. Goncharov constructed a series of examples of nonautostable models of finite algebraic dimension. Goncharov, Cholak, Shore, Khoussainov found

examples of nonautostable models with autostable enrichment of constants. Ash and Nerode established close connections between autostability and definability of hereditarily enumerable relations. Ash developed the method of labeling system that provided an essential advantage in the description of hereditarily arithmetic relations. This in turn, led to series of applications by Ash and Knight. Harizanov began important investigation of Turing spectra of relations.

Extensive studies of the lattice of r.e. substructures for a variety of recursive structures have been carried out. For example, there is a large body of work on the lattice of r.e. subspaces $\mathcal{L}(V_\infty)$ of an infinite dimensional recursive vector space by Ash, Bäuerle, Downey, Hird, Kalantari, Kurtz, Metakides, Nerode, Remmel, Retzlaff, Smith, Shore, Welch, and others. The lattices of r.e. subalgebras and r.e. ideals of a recursive Boolean algebra have been studied by Downey, Carroll, Goncharov, Morozov, Remmel, Yang and others. The lattice of r.e. open sets of a recursive topological space by Kalantari, Leggett, Retzlaff, Remmel, and others. Lattices of r.e. suborderings have been studied by Metakides, Remmel, Roy, and others. The lattices of r.e. affine spaces of a recursive vector space over an ordered field have been studied by Downey, Kalantari, Remmel and others. In addition, there has been significant work on providing a unified setting for the study of that lattice of r.e. substructures. For example, the study of the lattice of r.e. substructures of effective Steinitz systems, which cover both $L(V_\infty)$ and the lattice of r.e. algebraically closed subfields of an algebraically closed recursive field with an infinite transcendence basis, has been carried out by Baldwin, Downey, Metakides, Nerode, Remmel, and others. Remmel provided an even more general setting which covers all the lattices described above, see the article by Downey and Remmel in this volume. For a survey of results up to 1982, see the survey article by Nerode and Remmel [112].

For the fifth problem, Nurtazin proved the computability of the class of all constructive models of a fixed signature without function symbols and demonstrated that such a class is not computable if the signature contains functional symbols. Furthermore, Goncharov proved that the class of constructivizations for a decidable nonautostable model is not computable. A number of results concerning the complexity of index sets and the infinity of the Rogers semi-lattices of computable enumerations of classes of constructive models were obtained by Dobritsa. On the basis of the Ash method, the complexity of index sets in the arithmetic hierarchy was studied by Ash and Knight.

For the sixth type of problem see the survey articles by Gasarch for combinatorics, Downey for linear orderings, and especially the survey article by Cenzer and Remmel on Π_1^0 classes and Recursive Mathematics in this volume.

For results on the seventh type of problem, we refer the reader to the survey article on degrees of models by Knight in this volume.

Finally, prototypical results for the eighth type of problem include Feiner's result [43] that there is an r.e. Boolean algebra which is not isomorphic to any recursive Boolean algebra, and a recent result of Thurber [150] that every low_2 Boolean algebra is isomorphic to a recursive Boolean algebra. Also see the survey article by Downey on linear orderings in this volume for various results of this type for linear orders. There has been a significant number of papers on r.e. presented structures. For example, Ash, Downey, Goncharov, Jockusch, Knight, Metakides, Nerode, Remmel, Stob, Thurber, and others, have studied r.e. presented Boolean algebras; Ash, Downey, Goncharov, Jockusch, Knight, Metakides, Moses, Nerode, Remmel, Roy, Rosenstein, Soare, and others, have studied r.e. presented linear orderings; and Downey, Metakides, Nerode, and Remmel have studied r.e. presented vector spaces.

The list above is by no means complete nor is the list of contributors listed in each area complete. Nevertheless the reader will find that keeping these themes in mind will explain the motivation for many of the articles in this volume.

The questions above arose mainly in the study of Recursive Model Theory and Algebra. There is a large body of work in Recursive Analysis and Topology, as can be seen from the bibliography on Recursive Analysis and Topology at the end of this volume. The emphasis in the research on Recursive Analysis and Topology continues to be on analyzing the effective content of constructions and theorems of Analysis and Topology based on the definitions of computable real numbers, computable functions of the real numbers, and effectively closed sets.

Just as the formalization of computable functions in the 1930's naturally lead to Recursive Mathematics, the development of Complexity Theory by Blum, Cobham, Cook, Hartmanis, Karp, Ladner, Lewis, Sterns, and others in the 1960's and 1970's in Computer Science has led to development of a polynomial-time, and more generally, a feasible version of Recursive Mathematics which is called Polynomial-time or Feasible Mathematics. Of course, the definition of the polynomial-time hierarchy, the polynomial-space hierarchy, etc., were defined by analogy with the arithmetic hierarchy from

Computability Theory. The work of Karp on NP-complete problems has lead the ever expanding list of NP-complete problems, see Garey and Johnson [49], and to the increasing importance of the fundamental question of whether P equals NP . The paper by Baker, Gill, and Solovay [4] introduced oracle arguments into Complexity Theory. All these ideas, plus ideas from Recursive Mathematics, have played an important part in the development of Feasible Mathematics.

The work on Feasible Mathematics started with the work of Friedman and Ko [84, 46], where they developed a coherent notion of a polynomial-time computable function of the reals, and related a number of classical complexity-theoretic questions to questions about complexity of operations on polynomial computable functions. This subject has continued to develop (see the survey article by Ko in this volume). Remmel and Nerode developed a theory of the lattice of NP-substructures of a polynomial-time presented structure, and showed that priority arguments on oracles could play a fundamental role in the analysis of such lattices. The driving analogy in the Nerode-Remmel work is that recursive is to recursively enumerable as polynomial-time is to nondeterministic polynomial-time. Thus, for example, the polynomial-time analogue of Dekker's [30] result, that every r.e. subspace of a recursively presented infinite dimensional vector space over a recursive field has a recursive basis, is that every NP-subspace of a polynomial-time presented infinite dimensional vector space over a polynomial-time field has a basis in P . This analogy is true over certain infinite polynomial-time fields, but is oracle dependent over finite fields (see [115] and the article by Cenzer and Remmel on Complexity Theoretic Model Theory and Algebra in this volume).

Cenzer and Remmel have developed a rich theory of Polynomial-time and Feasible Model Theory which is also outlined in their paper in this volume. A number of interesting refinements of questions in Recursive Model Theory have arisen in Feasible Model Theory. For example, the questions of when a recursive model is isomorphic or recursively isomorphic to a polynomial-time model, or when a recursive model is isomorphic or recursively isomorphic to a polynomial-time model with a standard universe, such as the binary representation of the natural numbers or the unary representation of the natural numbers, leads to a surprisingly rich theory. Other developments in Feasible Mathematics include the work of Crossley, Nerode and Remmel [27, 116] on a polynomial-time analogue of Isol Theory which is far from a mere imitation of standard Isol Theory. Also other models of computation

have been studied. For example, Blum, Smale and Shub [8] developed a theory of complexity for computing with real numbers and Khoussainov and Nerode [77] have developed a theory of automata representable structures.

Overview of the Handbook

The Handbook of Recursive Mathematics contains over 1350 pages and hence we were forced to split the Handbook into two volumes. The editors decided that it was best to split the papers between the two volumes according to subject matter. The first volume covers Recursive Model Theory and the second volume covers Recursive Algebra, Analysis, Combinatorics and a variety of other topics. Since we did not originally plan for two volumes, the partition of the papers between the two volumes could not be accomplished without some overlap. That is, there are a number of papers in the second volume which are relevant to Recursive Model Theory and use recursive model theoretic techniques, and there are a number of papers in the first volume which are relevant to Recursive Algebra and use recursive algebraic techniques. We shall provide brief summaries of the papers in the two volumes below.

Volume 1: Recursive Model Theory

The first two papers of volume 1 were chosen because they provide good general introductions to Recursive or Computable Model Theory as developed in both the West, mainly the United States and Australia, and the East, mainly in the former Soviet Union. The rest of volume 1 is devoted to more specialized topics in Recursive Model Theory. The last paper of the volume is a survey paper by Cenzer and Remmel on Polynomial Time Model Theory and Algebra. The Cenzer and Remmel paper shows how seriously taking into account the resource bounds of computations greatly affects the types of questions and results that one obtains when considering effective content of model theoretic questions. It also surveys results on resource bounded versions of algebraic constructions, and thus provides a nice segue into the second volume.

In the first paper of volume 1, Harizanov provides a very valuable survey of results in Recursive or Computable Model Theory mainly from the Western perspective in her article “Pure Computable Model Theory”. In particular,

she supplies the model theoretic background as well as the proofs of a large number of basic results in computable model theory, including the basic effective completeness theorem and effective omitting types theorem, various results on conditions which ensure the existence of various types of decidable models including prime, homogeneous, and saturated models or models with effective sets of indiscernibles, results on decidable theories with only finitely many models or finitely many recursive models (the so-called Ehrenfeucht theories, or effective Ehrenfeucht theories), a theory of the degrees of models, and classification results on the number of computable models. Thus her article is an excellent place for a student who is interested in Recursive Model Theory as well as a valuable reference for established researchers in the field.

The article by Ershov and Goncharov, “Elementary Theories and Their Constructive Models”, is a good introductory article for those who are unfamiliar with the approach of the Ershov school to Recursive Model Theory. They provide a survey of some of the basic existence theorems for constructive and strongly constructive models. In particular they provide a proof of the Goncharov-Peretyat'kin criterion for the existence of a decidable homogeneous model. They also apply their theory to Boolean algebras, which is a rich source of examples and for which a well developed theory of constructive models exists.

The paper, “Isomorphic Recursive Structures” by Ash, surveys several results on when a recursive structure is unique up to recursive isomorphisms, Δ_2^0 isomorphisms, etc.. In a similar spirit, Ash looks at conditions which ensure that a given relation is always recursive, r.e., Σ_2^0 , etc., in any recursive structure which is isomorphic to the original recursive structure. Such relations are called intrinsically recursive, r.e., etc.. A number of characterizations of intrinsically recursive and r.e. relations can be found in the Ash article.

The article “Computable Classes and Constructive Models” by Dobritsa presents a large body of results on when the set of recursive models, which are isomorphic to a given recursive model or are extensions of a given recursive model, are computable; in the sense that one can effectively list all such models. In addition, he surveys a number of results on conditions which ensure that the class of such recursive models lies in the arithmetic hierarchy. A theory of effective reductions of one class of models to another class of models is also presented.

In the paper “ Σ -Definability of Algebraic Systems”, Ershov proves a number of interesting results on the extension of his theory of numerations and constructivizations where recursive and computable functions are replaced by definability notions from the theory of admissible sets. He thus defines the notion of a Σ -definable algebraic structure of an admissible set, and proves a number of analogues of results on constructivizations of algebraic structures in this setting. This work opens up another extension of Recursive Mathematics using notions from α -recursion theory and admissible sets.

In the article “Autostable Models and Algorithmic Dimensions”, Goncharov defines several reducibilities which can be defined on the class of constructivizations, $Con(\mathfrak{M})$, of a given model $\mathfrak{M}$. That is, given two constructivizations $\nu, \mu : \omega \rightarrow \mathfrak{M}$, we say that

1. $\nu \leq_K \mu$, if there exists a recursive function f such that $\nu = \mu f$ (Kolmogorov reducibility),
2. $\nu \leq \mu$, if there is an automorphism φ of $\mathfrak{M}$ such that $\varphi\nu \leq_K \mu$ (autoreducibility),
3. $\nu \leq_U \mu$, if there exists a computable operator F such that $F(\chi_{\mu^{-1}(S)}) = \chi_{\nu^{-1}(S)}$ for all relations S which are stable in the sense that S is invariant under automorphisms of $\mathfrak{M}$ (uniform reducibility),
4. $\nu \leq_P \mu$, if there is a partial recursive function f such that if φ_n is the characteristic function of $\mu^{-1}(S)$ for some stable relation S , then $\varphi_{f(n)}$ is the characteristic function of $\nu^{-1}(S)$ (program reducibility), and
5. $\nu \leq_{Alg} \mu$, if every stable relation of $\mathfrak{M}$ which is decidable under the constructivization μ is also decidable under the constructivization ν (algebraic reducibility).

Goncharov then discusses the relations between these reducibilities, and surveys a number of results on conditions which ensure that the set of reducibility classes of a model has cardinality 1 , ω , or is finite.

In her article “Degrees of Models”, Knight presents another significant theme in recursive model theory, namely, she surveys a number of results on the sets of degrees of structures which are isomorphic to a given structure or the set of degrees of models of a particular theory. For example, let $DI(\mathcal{A}) = \{\deg(\mathcal{B}) : \mathcal{B} \text{ is isomorphic to } \mathcal{A}\}$. Generally $DI(\mathcal{A})$ is closed upwards so that

it is natural to ask whether there is a least element in $DI(\mathcal{A})$, i.e., when is there a degree $\mathbf{c}$ such that $DI(\mathcal{A}) = \{\mathbf{d} : \mathbf{d} \geq_T \mathbf{c}\}$? Knight surveys a number of results on structures for which such a degree $\mathbf{c}$ exists, and on structures for which no such degree $\mathbf{c}$ exists. More generally, one can define a structure to have α -th jump degree $\mathbf{d}$ for a given recursive ordinal α if $\{\mathbf{b}^{(\alpha)} : \mathbf{b} \in DI(\mathcal{A})\} = \{\mathbf{c} : \mathbf{c} \geq_T \mathbf{d}\}$, where $\mathbf{b}^{(\alpha)}$ denotes the α -th jump of $\mathbf{b}$. Knight also surveys a number of interesting results on structures which have an α -th jump degree. Finally Knight presents results on the possible degrees of non-standard models of arithmetic and on results which guarantee the existence of recursive models of nonrecursive theories.

In his article “Groups of Computable Automorphisms”, Morozov provides a survey of results on the groups of all automorphisms, of all recursive automorphisms, of all arithmetic automorphisms, etc., of a recursive or constructible model over an effective language. He starts his survey with a number of general results on the set of recursive automorphisms of a recursive model. For example, there exist many examples of recursive models which have $2^{\aleph_0}$ automorphisms but only a single recursive automorphism. In fact, one can construct a homogeneous strongly constructive model which has $2^{\aleph_0}$ automorphisms, but every constructivization of that model has only one recursive automorphism. Similarly, one can construct a hyperarithmetic model with $2^{\aleph_0}$ automorphisms, but which has a unique hyperarithmetical automorphism. He also surveys results about when a recursive model $\mathfrak{M}$ can have a computable set of recursive automorphisms $\text{Aut}_R(\mathfrak{M})$. He ends his survey with a number of interesting results on the set of recursive automorphisms of recursive Boolean algebras and vector spaces. For example, it is known that if B is a decidable atomic Boolean algebra and B' is any recursive Boolean algebra, then the fact that $\text{Aut}_r(B)$ is isomorphic to $\text{Aut}_r(B')$ implies that B is isomorphic to B' . On the other hand, a result of Rempel shows that for any recursive Boolean algebra with infinitely many atoms, there exists a recursive Boolean algebra C isomorphic to B such that every recursive automorphism of C moves only finitely many atoms. Morozov’s article shows that there is a surprisingly rich theory of the various automorphism groups of recursive models.

In his article “Constructive Models and Finitely Axiomatizable Theories”, Peretyat’kin surveys a number of results on finitely axiomatizable theories. Finitely axiomatizable theories have been extensively studied in model theory, and Peretyat’kin provides a nice summary of such results in his article.

He then surveys results on the complexity of the the prime, countably homogeneous, and countable saturated models of such theories as well as a large number of index set results for finitely axiomatizable theories. He also surveys a number of interesting results on the complexity of the Lindenbaum algebra of finitely axiomatizable theories. Finally he presents a long list of open problems in the area.

As mentioned above, the last article of volume 1 looks at how resource bounds affect questions in Recursive Model Theory. That is, one restricts one's attention to models where the underlying universe, functions, and relations are limited to be in some natural complexity class such as polynomial-time or polynomial-space, and studies how such restrictions affect the types of results developed in Recursive Model Theory. As the reader can see from the article by Cenzer and Remmel in this volume, and the article by Ko on Polynomial Time Analysis in the second volume, restricting one's attention to feasible functions does not produce a theory which is a mere imitation of results in Recursive Mathematics. Cenzer and Remmel provide an introduction to the types of questions which arise in Feasible Mathematics in their article "Complexity Theoretic Model Theory and Algebra". Cenzer and Remmel give a survey of their theory of polynomial-time and feasible models in this paper. They also survey the theory of Polynomial-time Algebra and the theory of the lattices of NP-substructures of a polynomial-time structure that has been developed by Nerode and Remmel. This paper provides a good introduction to an area that is really in its infancy. Nevertheless, the results achieved so far show that this is an interesting area of research in which a number of new phenomena arise which do not appear in Recursive Algebra. Thus Polynomial-time Model Theory and Algebra offer a rich opportunity for further research.

Finally, volume 1 ends with two extremely valuable bibliographies. The first is a bibliography on Recursive Algebra which was compiled by Kalantari. The second is a bibliography on Recursive Analysis and Topology that was compiled by Kalantari and Brattka. Both of these bibliographies were put together in response to a rather late request by the editors. Because of the lack of time, the bibliographies could not be as exhaustive as desired and we apologize to those whose work has been inadvertently omitted. Nevertheless, these bibliographies are valuable additions to this volume and provide those researchers and students who want a deeper treatment than is presented in this volume a valuable guide to the literature.

Volume 2: Recursive Algebra, Analysis, and Combinatorics

In the first paper of the second volume, “ Π_1^0 Classes in Mathematics”, Cenzer and Remmel provide a fairly comprehensive survey of the uses of Π_1^0 classes in Recursive Mathematics. It is well known to recursion theorists that Π_1^0 classes are ubiquitous in many areas of mathematics. The reason why Π_1^0 classes arise so naturally in Recursive Mathematics is because it is often the case that the set of solutions to a recursive problem can be viewed as the set of paths through a recursive tree. For example, the set of proper k -colorings of a recursive graph G can be viewed as the set of paths through a recursive tree T_G , and hence is a Π_1^0 class. The question then becomes whether for every recursive tree T , the set of paths through T is in one-to-one degree preserving correspondence with the set of k -colorings of some recursive graph. If so, then one can transfer a vast number of results on the possible degrees of elements of Π_1^0 classes and index set results for Π_1^0 classes to results about the set of k -colorings of a recursive graph. Cenzer and Remmel give a large number of problems for which this type of correspondence, and weaker correspondences, exist. They also give an overview of basic results on Π_1^0 classes, and explain how such results can be used to give a complexity analysis of a large number of problems considered in Recursive Mathematics.

Downey, in his paper “Recursion Theory and Linear Orderings”, provides an extensive survey of results on recursive orderings. A number of very interesting questions have arisen in the theory of recursive orderings, including the question of classifying when certain recursive orderings are unique up to recursive isomorphisms, the question of finding the effective dimension of a recursive partial ordering, the question of when an r.e. presented linear ordering or low linear ordering is isomorphic to a recursive linear ordering, the question of when a linear ordering has an effective ω or ω^* sequence, and many others. Downey’s paper is an excellent starting point for those who are not conversant with Recursive Mathematics, as his paper starts out with a good introduction to many basic theorems and techniques of modern computability theory.

In the paper, “Effective Algebras and Closure Systems: Coding Properties”, Downey and Remmel show that a large number of results on the lattice of r.e. substructures of various recursive structures can be given uniform proofs which often involve simple coding arguments based on results from the lattice of r.e. sets. They work in a general setting due to Remmel [129]

called effective closure systems, which cover a large number of lattices, including the lattice of r.e. sets, the lattice of r.e. subspaces of an infinite dimensional recursive vector space over a recursive field, the lattice of r.e. algebraically closed subfields of a recursive algebraically closed field with an infinite transcendence basis, and the lattice of r.e. ideals and the lattice of r.e. subalgebras of a recursive Boolean algebra. They also give some examples of lattices which cannot be covered by their general setting.

Gasarch provides an extensive survey of results on the effective content of theorems in combinatorics in his article "Recursive Combinatorics". Many results in infinite combinatorics are not effective, and there is a large body of work on classifying which of the various theorems from graph theory, orderings, Ramsey theory, matching theory, etc., are or are not effective. Moreover, there are many beautiful recursive variations of combinatorial problems that provide many avenues of interesting research. For example, Dilworth's theorem that every partial ordering of width n can be covered by n chains is not effective. Indeed, it is relatively easy to construct recursive partial orders of width 2 which cannot be covered by 2 recursive chains. However Kierstead [78] showed that every recursive partial order of width n could be covered by $\frac{1}{4}(5^n - 1)$ recursive chains. The exact bound on how many recursive chains are required to cover a recursive partial order of width n is not known. Gasarch's article has extensive connections with the articles of Downey on recursive orderings, of Cenzer and Remmel on Π_1^0 classes and Recursive Mathematics, and of Kierstead on recursive and on-line colorings.

In his article "Constructive Abelian Groups", Khisamiev looks at the effective content of the theory of Abelian groups. First he surveys a large number of results on various classes of groups which have a constructivization, i.e., groups which have a recursive presentation, and groups which have a strong constructivization, i.e., groups which have a decidable presentation. This provides a number of nice examples of the differences between constructivizations and strong constructivizations. He then goes on to classify which groups have a unique recursive presentation up to recursive isomorphisms. He also presents a number of results which show that Ulm's theorem for p -groups is not effective. Finally he surveys results on the constructibility of torsion-free abelian groups and results on the constructibility of subgroups and factor groups in constructible groups. The Khisamiev article thus provides a fascinating chapter of the interactions of Recursive Mathematics with a classical Algebra.

The article by Kierstead, “Recursive and On-Line Graph Coloring”, studies another very interesting connection of recursive mathematics with computer science, namely the theory of on-line algorithms. In the on-line coloring problem for a graph G , the set of vertices of the graph are presented one at a time and the *on-line* algorithm is required to decide the color of that vertex based only on the knowledge of the colors assigned to, and the edges between, the points which have previously appeared plus the current vertex. The connection with finding recursive colorings is due to the fact that to give a procedure to recursively color a graph, one can only use local information, and hence an on-line coloring algorithm can be translated into results about recursive colorings. It turns out that many of the techniques developed in the study of recursive colorings of recursive graphs have applications to the on-line coloring of graphs. In particular, certain recursive counterexamples can be translated to give counterexamples for on-line colorings. It should be noted that the theory of on-line colorings contains many subtle questions which have no analogues in recursive graph theory, and Kierstead’s article presents a number of such questions and surveys a number of interesting results in this area.

Ko presents a survey of results in Feasible Analysis in his article “Polynomial Time Analysis”. After a brief survey of results on recursive analysis and the basic definitions of the complexity theoretic hierarchies, Ko presents the Friedman-Ko model of complexity of computable functions $f : [0, 1] \rightarrow \mathbb{R}$. He then studies the complexity of various operations on computable real functions including maximization, root-finding, integration, solving ordinary differential equations, and solving integral equations. In each of these cases, there are very strong connections with standard Complexity Theory. For example, one of the first results in this area is due to Friedman, and states that the following are equivalent:

1. $P = NP$.
2. For each polynomial-time computable function $f : [0, 1]^2 \rightarrow \mathbb{R}$, the function $g(x) = \max\{f(x, y) : 0 \leq y \leq 1\}$ is polynomial-time computable.
3. For each polynomial-time computable function $f : [0, 1] \rightarrow \mathbb{R}$, the function $g(x) = \max\{f(y) : 0 \leq y \leq x\}$ is polynomial-time computable.
4. For each polynomial-time computable function $f : [0, 1] \rightarrow \mathbb{R}$ that is infinitely differentiable, the function $g(x) = \max\{f(x, y) : 0 \leq y \leq 1\}$ is polynomial-time computable.

For each of the operations listed above, Ko surveys results on conditions which guarantee polynomial-time or polynomial-space solutions, as well as results connecting the problems of finding polynomial-time or polynomial-space solutions in general with well known separation problems from Complexity Theory. Thus, Ko presents another interesting area of Feasible Mathematics which is a fruitful area for further research.

The theory of recursive or constructive Boolean algebras provides an immensely rich area for Recursive Mathematics, see for example Remmel's survey article [130] and Goncharov's book [57]. In his article, "Generally Constructive Boolean Algebras", Odintsov surveys a number of results on extensions of constructible Boolean algebras. A strongly constructive Boolean algebra is a Boolean algebra B such that there exists a map $\nu : \omega \rightarrow B$ such that the set of all $\langle n, b_0, \dots, b_{n-1} \rangle$, where n is an index of a first order formula φ_n such that $B \vdash \varphi_n(\nu(b_0), \dots, \nu(b_{n-1}))$, is recursive. One can generalize this notion in several ways. For example, one could restrict the set of formulas φ_n to consists only of Σ_n formulas or Π_n formulas. Similarly, one could insist that the set of all $\langle n, b_0, \dots, b_{n-1} \rangle$, where n is an index of a first order formula φ_n such that $B \vdash \varphi_n(\nu(b_0), \dots, \nu(b_{n-1}))$, be arithmetic rather than recursive. One could also consider formulas in some fragment of second order logic like $L(Q)$, where Q is the Mostowski quantifier, "there exist infinitely many". Finally one can add extra predicates such as predicates for various types of filters. Odintsov surveys results on all of these possible extensions and shows that there are many natural extensions of the basic notions of constructivizability and strong constructivizability which yield fruitful areas of study.

The article "Reverse Algebra", by Simpson and Rao, provides an introduction and survey of results on Reverse Mathematics and Algebra. The goal of Reverse Mathematics is to answer the question about which set existence axioms are needed to prove theorems in ordinary mathematics. To this end, five subsystems of second order arithmetic have been introduced: RCA_0 which consists of the usual axioms for the ordered semi-ring $(\omega, +, \cdot, 0, 1, <)$ plus Δ_1^0 -comprehension and Σ_1^0 -induction, WKL_0 which is RCA_0 plus the statement of weak König's Lemma (every infinite n -ary branching tree has an infinite path), ACA_0 which is RCA_0 plus comprehension axioms for all arithmetical formulas, ATR_0 which includes ACA_0 plus the axiom that arithmetical comprehension can be iterated along any countable well ordering, and $\Pi_1^1\text{-CA}$ which consists of RCA_0 plus comprehension axioms for all Π_1^1 formulas. At

this point, there are a large number of theorems of classical mathematics that have been classified according to these five subtheories. For example, consider the theorem that every commutative ring has a maximal ideal. Friedman, Simpson, and Smith [47] have shown that this theorem can be proved in ACA_0 , and moreover that any theorem which can be proved in ACA_0 can be proved in the theory RCA_0 plus the axiom that every commutative ring has a maximal ideal. Thus one could say that this theorem has the same power as arithmetic comprehension. Simpson and Rao survey a number of theorems in Algebra that have been classified with this framework and provide an interesting section on the connections between Reverse Mathematics and Recursive Mathematics.

Relations with other subjects and future developments

We have attempted to show in this introduction that there is a natural relation between Recursive Mathematics and other areas of Mathematics. For example, there is a natural relation between Recursive Mathematics and Constructive Mathematics. There is also an intimate relation between Recursive Mathematics and Reverse Mathematics which was started by Friedman to classify theorems of mathematics by their proof theoretic strength over weak theories of arithmetics (see Simpson's article in this volume). Similarly, some of the work on Recursive Combinatorics is naturally connected with the theory of on-line algorithms (see Kierstead's article in this volume).

There are many important unsolved problems in Recursive Mathematics and many areas which require further development. For example, the effectiveness of model-theoretic constructions continues to supply a rich source of problems. Pour-El and Richards [122] looked at the effective content of results from mathematical physics, but their work has just scratched the surface of a large area of interesting problems concerning the effective content of other applied areas, such as physics, control theory (see the article by Ge and Nerode [50]), and statistics. The work on Feasible Model Theory still lacks the sort of general sufficient conditions for the construction of polynomial-time models that abound in Recursive Model Theory. Similarly a theory of Polynomial-space Model Theory and Algebra is yet to be developed. In general, the work on Feasible Algebra has not yet yielded the equivalences

with classical complexity theoretic questions that can be found in the theory of Polynomial-time Analysis. In contrast, the use of the priority method which has many applications in Recursive Model Theory and Algebra has not found any uses in Recursive Analysis. This raises the question of whether there is an essential difference between Recursive Model Theory and Algebra versus Recursive Analysis, or whether we have just not studied sufficiently deep questions in Recursive Analysis to require the priority method at this time. Many open questions remain on the effective content of fields and topological vector spaces.

The exact relation between Recursive Mathematics and other areas is not well understood. For example, are there relations between Recursive Mathematics and typed lambda calculus interpretations of higher order intuitionistic logic, as in the work of Girard, or between Recursive Mathematics and untyped lambda calculus models of intuitionistic Zermelo-Fraenkel set theory, as in the work of McCarty [91]? In the same vein, the relation between Friedman-Simpson's weak systems which have only limited comprehension, the systems of Reverse Mathematics, and Recursive Mathematics is still not completely understood. For example, does the finer analysis of the effective content of König's lemma, i.e., the full theory of Π_1^0 classes, give a finer analysis of proof theoretic strengths? For example, can Reverse Mathematics distinguish between problems which can represent an arbitrary recursively bounded Π_1^0 class as opposed to the class of separating sets of a pair of r.e. sets (see the article by Cenzer and Remmel on Π_1^0 classes in this volume)? We hope that the methods surveyed in this book will help to clarify these matters.

Acknowledgments

There are number of people who deserve special thanks for this volume. First we must thank all the authors who contributed articles. Our hope was to have a volume which would first exhibit the beauty and depth of the subject and second inspire a new generation of researchers to go into the field of Recursive Mathematics. Only time will tell if the latter goal will be met but we are confident that reader will agree that our first goal has been achieved splendidly. Many of the authors also refereed articles for the volume. A very special note of thanks must go to Victor Marek who took over the job of coordinating the volume when the last editor was on leave in industry for

the last two years. Without Victor's tireless efforts on behalf of the volume, it would not have been possible to produce it this year.

We are grateful to numerous colleagues who helped us edit this volume. Our $\text{\LaTeX}$ wizard, Professor Graeme Bailey of Cornell University, helped us transform the variety of styles supplied by authors into a coherent whole. Our thanks go to a number of young researchers at Cornell University, Jennifer Davoren, Dennis Hirschfeldt, Robert Milnikel, David Solomon, and Anke Walz, who assisted the authors in Kazakhstan and in Russia to improve the readability and linguistic correctness of their contributions. We are grateful to Diana Drake who provided administrative support at Cornell. When the electronic connections between the United States and Novosibirsk failed, we were assisted by Professor Sergiey Artemov (of Moscow State University and Cornell University) and Professor Lev Beklemishev (of Moscow State University) who helped to provide the required connections often via the telephone. Finally a special thanks is due to Julia Knight who edited the contribution by the late Christopher Ash.

We gratefully acknowledge support for our work on the volume. In particular, J. Remmel has been partially supported by US Department of Commerce agreement 70-NANB5H1164 and NSF grant DMS-9306427. V. Marek has been partially supported by NSF grant IRI-9619233 and USARO contract DAAH-04-96-1-0398. A. Nerode and his collaborators at Cornell University have been partially supported by USARO MURI DAAH-04-96-10341, Center for Foundations of Intelligent Systems at Cornell University.

References

- [1] O. Aberth, Analysis in the computable number field, *J. Assoc. Comput. Mach.*, **15** (1968) 275–299.
- [2] C. H. Applebaum, Isomorphisms of ω -groups, *Notre Dame J. Formal Logic*, **12** (1971) 238–248.
- [3] E. Artin and O. Schreier, Algebraische Konstruktion reeller Körper, *Abh. Math. Sem. Univ. Hamburg*, **5** (1927) 85–99.
- [4] T. Baker, J. Gill and R. Solovay, Relativizations of the $P = ? NP$ question, *SIAM J. Comput.*, **4** (1975) 431–442.

- [5] P. Bernays and D. Hilbert, *Grundlagen der Mathematik*, Vol. I, II, (1934, 1939), *Grundlehren Math. Wiss.*; [2nd. edn. (1968, 1970)].
- [6] E. Bishop, *Foundations of Constructive Analysis*, (McGraw-Hill, New York, Toronto, London, 1967).
- [7] E. Bishop and D. S. Bridges, *Constructive Analysis*, *Grundlehren Math. Wiss.*, **279** (1985).
- [8] L. Blum, M. Shub and S. Smale, On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines, *Bull. Amer. Math. Soc. (N.S.)*, **21** (1989) 1–46.
- [9] M. Blum, A machine independent theory of the complexity of recursive functions, *J. Assoc. Comput. Mach.*, **14** (1967) 322–336.
- [10] W. W. Boone, Certain simple unsolvable problems in group theory, I–VI, *Nederl. Akad. Wetensch. Proc. Ser. A*, **57** (1954) 231–237, 492–497; **58** (1955) 252–256, 571–577; **60** (1957) 22–27, 227–232.
- [11] L. E. J. Brouwer, *Collected Works*, Vol. 1, *Philosophy and Foundations of Mathematics*, A. Heyting, (ed.), (North-Holland, Amsterdam, Oxford, 1975).
- [12] L. E. J. Brouwer, *Collected Works*, Vol. 2, *Geometry, Analysis, Topology and Mechanics*, H. Freudenthal, (ed.), (North-Holland, Amsterdam, Oxford, 1976).
- [13] L. E. J. Brouwer and B. de Loor, *Intuitionistischer Beweis des Fundamentalsatzes der Algebra*, *Koninklijke Nederl. Akad. Wetensch. Proc.*, **27** (1924) 186–188.
- [14] G. S. Ceitin, Algorithmic operators in constructive complete separable metric spaces (Russian), *Dokl. Akad. Nauk SSSR*, **128** (1959) 49–52.
- [15] D. Cenzer and J. B. Remmel, Polynomial-time versus recursive models, *Ann. Pure Appl. Logic*, **54** (1991) 17–58.
- [16] D. Cenzer and J. B. Remmel, Feasible graphs and colorings, *Math. Logic Quart.*, **41** (1992) 327–352.
- [17] D. Cenzer and J. B. Remmel, Polynomial-time abelian groups, *Ann. Pure Appl. Logic*, **56** (1992) 313–363.

- [18] C. C. Chang and H. J. Keisler, *Model Theory*, 3rd. edn., Stud. Logic Found. Math., **73** (1990); [1st. edn. 1973, 2nd. edn. 1977].
- [19] A. Church, An unsolvable problem of elementary number theory, *Amer. J. Math.*, **58** (1936) 345–363.
- [20] A. Church and S. C. Kleene, Formal definitions in the theory of ordinal numbers, *Fund. Math.*, **28** (1937) 11–21.
- [21] A. Cobham, The intrinsic computational difficulty of functions, in: *Proc. Inter. Cong. for Logic, Methodology, and Philos. Sci.*, (Jerusalem, 1984), North Holland, (1965) 24–30.
- [22] S. A. Cook, The complexity of theorem proving procedures, *ACM Sympos. Theory of Computation*, (1971) 151–158.
- [23] J. N. Crossley, *Constructive Order Types*, (North-Holland, Amsterdam, 1969).
- [24] J. N. Crossley, Recursive equivalence, *Bull. London Math. Soc.*, **2** (1970) 129–151.
- [25] J. N. Crossley and A. Nerode, Combinatorial Functors, *Ergeb. Math. Grenzgeb. Ser. 3*, **81** (1974).
- [26] J. N. Crossley and A. Nerode, Effective dimension, *J. Algebra*, **41** (1976) 398–412.
- [27] J. N. Crossley and A. Nerode, Cancellation laws for polynomial-time p -isolated sets, *Ann. Pure Appl. Logic*, **56** (1992) 147–172.
- [28] M. D. Davis, Yu. Matijasevič and J. Robinson, Hilbert's tenth problem. Diophantine equations: positive aspects of a negative solution, in: *Mathematical Developments Arising from Hilbert Problems*, (N. Illinois Univ., De Kalb, Ill, 1974), *Proc. Sympos. Pure Math.*, **28** (1976) 323–378.
- [29] M. D. Davis, H. Putnam and J. Robinson, The decision problem for exponential diophantine equations, *Ann. of Math.*, Ser. 2, **74** (1961) 425–436.
- [30] J. C. E. Dekker, Countable vector spaces with recursive operations. Part I, *J. Symbolic Logic*, **34** (1969) 363–387.
- [31] J. C. E. Dekker, Countable vector spaces with recursive operations. Part II, *J. Symbolic Logic*, **36** (1971) 477–493.

- [32] J. C. E. Dekker, Recursive equivalence types and cubes, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981) 87–121.
- [33] J. C. E. Dekker and J. Myhill, Recursive equivalence types, *Publ. Math. Univ. California*, **3** (1960) 67–214.
- [34] E. Ellentuck, The first order properties of Dedekind finite integers, *Fund. Math.*, **63** (1968) 7–25.
- [35] E. Ellentuck, Nonrecursive combinatorial functions, *J. Symbolic Logic*, **37** (1972) 90–95.
- [36] E. Ellentuck, Sylow subgroups of a regressive group, *Houston J. Math.*, **5** (1979) 46–67.
- [37] Yu. L. Ershov, Constructive models (Russian), in: *Izbr. Vopr. Alg. i Log.*, [Selected Questions in Algebra and Logic], *Sbornik posvjascen. pamjati A. I. Mal'ceva*, [A collection dedicated to the memory of A. I. Mal'tsev], Izdat. Nauk Sibirsk. Otdel., Novosibirsk, (1973) 111–130.
- [38] Yu. L. Ershov, Skolem functions and constructive models (Russian), *Algebra i Logika*, **12** (1973) 644–654, 735; [translated in: *Algebra and Logic*, **12** (1973) 368–373].
- [39] Yu. L. Ershov, Theory of Numerations III (Constructive Models) (Russian), *Lib. Dept. Algebra and Math. Logic Novosibirsk Univ.*, **13**, (Novosibirsk Gosudarstr. Univ., Novosibirsk, 1974).
- [40] Yu. L. Ershov, Hereditarily effective operations (Russian), *Algebra i Logika*, **15** (1976) 642–654. [translated in: *Algebra and Logic*, **15** (1976) 400–409.
- [41] Yu. L. Ershov, Theory of Numerations (Russian), *Monographs in Math. Logic and Foundations of Math.*, (Nauka, Moskva, 1977).
- [42] Yu. L. Ershov, *Theorie der Numerierungen, III*, (translated from Russian and edited by G. Asser and H.-D. Hecker), *Z. Math. Logik Grundlag. Math.*, **23** (1977) 289–371.
- [43] L. J. Feiner, Hierarchies of Boolean algebras, *J. Symbolic Logic*, **35** (1970) 365–374.

- [44] R. M. Friedberg, Two recursively enumerable sets of incomparable degrees of unsolvability, *Proc. Nat. Acad. Sci. U.S.A.*, **43** (1957) 236–238.
- [45] H. M. Friedman, Set theoretic foundations for constructive analysis, *Ann. of Math.*, Ser. 2, **105** (1977) 1–28.
- [46] H. M. Friedman, The computational complexity of maximization and integration, *Adv. Math.*, **53** (1984) 80–98.
- [47] H. M. Friedman, S. G. Simpson and R. L. Smith, Countable algebra and set existence axioms, *Ann. Pure Appl. Logic*, **25** (1983) 141–181; addendum, *ibid.* **27** (1985) 319–320.
- [48] A. Fröhlich and J. C. Shepherdson, Effective procedures in field theory, *Philos. Trans. Roy. Soc. London*, Ser. A, **248** (1956) 407–432.
- [49] M. R. Garey and D. S. Johnson, *Computers and Intractability, a guide to the theory of NP-completeness*, (W. H. Freeman, San Francisco, 1979).
- [50] X. Ge and A. Nerode, Effective content of the calculus of variations I: semi-continuity and the chattering lemma, *Ann. Pure Appl. Logic*, **78** (1996) 127–146.
- [51] K. Gödel, Über formal unentscheidbare Sätze der *Principia mathematica* und verwandter Systeme I, *Monatsh. Math. Phys.*, **38** (1931) 173–198; [translated as: On formally undecidable propositions of *Principia Mathematica* and related systems I, (translated by B. Meltzer, with an introduction by R. B. Braithwaite), (Basic Books, NY, 1963, reprinted Dover, NY, 1992); also in: *From Frege to Gödel: A Source Book in Logic, 1879–1931*, J. van Heijenoort, (ed.), (Harvard Univ. Press, Cambridge, Mass. and Oxford Univ. Press, London, 1967) 592–617.]
- [52] S. S. Goncharov, The constructivizability of superatomic Boolean algebras (Russian), *Algebra i Logika*, **12** (1973) 31–40, 120; [translated in: *Algebra and Logic*, **12** (1973) 17–22].
- [53] S. S. Goncharov, Autostability and computable families of constructivizations (Russian), *Algebra i Logika*, **14** (1975) 647–680; [translated in: *Algebra and Logic*, **14** (1975) 392–409].
- [54] S. S. Goncharov, Certain properties of the constructivization of Boolean algebras (Russian), *Sibirsk. Math. Zh.*, **16** (1975) 264–278; [translated in: *Siberian Math. J.*, **16** (1975) 203–214].

- [55] S. S. Goncharov, Constructive models of $\aleph_1$ -categorical theories (Russian), *Mat. Zametki*, **23** (1978) 885–888; [translated in: *Math. Notes*, **23** (1978) 486–487].
- [56] S. S. Goncharov, Strong constructivizability of homogeneous models (Russian), *Algebra i Logika*, **17** (1978) 363–388, 490; [translated in: *Algebra and Logic*, **17** (1978) 247–263].
- [57] S. S. Goncharov, Countable Boolean Algebras (Russian), (Nauka. Sibirsk. Otdel., Novosibirsk, 1988).
- [58] R. L. Goodstein, Recursive analysis, in: *Constructivity in Mathematics*, (Proc. Colloq., Amsterdam, Aug. 26–31, 1957), A. Heyting, (ed.), *Stud. Logic Found. Math.*, (1959) 37–42.
- [59] L. Harrington, Recursively presentable prime models, *J. Symbolic Logic*, **39** (1974) 305–309.
- [60] J. Hartmanis, On sparse sets in $NP - P$, *Inform. Process. Lett.*, **16** (1983) 55–60.
- [61] J. Hartmanis and R. E. Stearns, On the computational complexity of algorithms, *Trans. Amer. Math. Soc.*, **117** (1965) 285–306.
- [62] M. J. Hassett, Recursive equivalence types and groups, *J. Symbolic Logic*, **34** (1969) 13–20.
- [63] J. Hauck, Funktional-Rekursion, *Z. Math. Logik Grundlag. Math.*, **18** (1972) 31–36.
- [64] L. S. Hay, The co-simple isols, *Ann. of Math.*, Ser. 2, **83** (1966) 231–256.
- [65] A. Heyting, Die formalen Regeln der intuitionistischen Logik, *Sitz. Preuss. Akad. Wiss. Phys.-Math. Kl.*, (1930) 42–56; Die formalen Regeln der intuitionistischen Mathematik, *ibid.*, (1930) 57–71; Die formalen Regeln der intuitionistischen Mathematik III, *ibid.*, (1930) 143, 158–169.
- [66] A. Heyting, *Intuitionism, An Introduction*, (North-Holland, Amsterdam, 1956); [2nd. rev. edn., 1966].
- [67] A. G. Higman, Subgroups of finitely presented groups, *Proc. Roy. Soc. London Ser. A*, **262** (1961) 455–475.+

- [68] D. Hilbert, Mathematische Probleme, *Archiv. f. Math. und Phys.*, Ser. 3, **1** (1901) 44–63, 213–237.
- [69] D. Hilbert and W. Ackermann, *Principles of Mathematical Logic*, (translated from German by L. M. Hammond, G. G. Lechie and F. Steinhardt, edited with notes by R. E. Luce), (Chelsea, 1950).
- [70] S. Homer and W. Maass, Oracle-dependent properties of the lattice of NP sets, *Theoret. Comput. Sci.*, **24** (1983) 279–289.
- [71] J. E. Hopcroft and J. D. Ullman, *Formal Languages and their Relations to Automata*, (Addison-Wesley, Reading, Mass, 1969).
- [72] C. G. Jockusch Jr., Ramsey's theorem and recursion theory, *J. Symbolic Logic*, **37** (1972) 268–280.
- [73] C. G. Jockusch Jr. and R. I. Soare, Degrees of members of Π_1^0 classes, *Pacific J. Math.*, **40** (1972) 605–616.
- [74] C. G. Jockusch Jr. and R. I. Soare, Π_1^0 classes and degrees of theories, *Trans. Amer. Math. Soc.*, **173** (1972) 33–56.
- [75] I. Kalantari and A. Retzlaff, Maximal vector spaces under automorphisms of the lattice of recursively enumerable vector spaces, *J. Symbolic Logic*, **42** (1977) 481–491.
- [76] R. M. Karp, Reducibilities among combinatorial problems, in: *Complexity of Computer Computations*, (Proc. Sympos. IBM Watson Res. Ctr., Yorktown Heights, NY, Mar. 20–22, 1972), R. E. Miller, J. W. Thatcher and J. D. Bohlinger, (eds.), (Plenum Press, New York, 1972) 85–103.
- [77] B. M. Khoussainov and A. Nerode, Automatic presentations of structures, in: *Logic and Computational Complexity*, (Papers from Internatl. Workshop, LCC '94, Oct. 13–16, 1994, Indianapolis, IN), D. Leivant (ed.), *Lecture Notes in Comput. Sci.*, **960** (1995) 367–392.
- [78] H. A. Kierstead, An effective version of Dilworth's theorem, *Trans. Amer. Math. Soc.*, **268** (1981) 63–77.
- [79] S. C. Kleene, λ -definability and recursiveness, *Duke Math. J.*, **2** (1936) 340–353.
- [80] S. C. Kleene, Recursive predicates and quantifiers, *Trans. Amer. Math. Soc.*, **53** (1943) 41–73.

- [81] S. C. Kleene, Introduction to Metamathematics, (D. Van Nostrand, Princeton, 1952, and Elsevier, Amsterdam, 1964, 1971).
- [82] S. C. Kleene, Hierarchies of number-theoretic predicates, *Bull. Amer. Math. Soc.*, **61** (1955) 193–213.
- [83] K. Ko, The maximum value problem and NP real numbers, *J. Comput. System Sci.*, **24** (1982) 15–35.
- [84] K. Ko and H. M. Friedman, Computational complexity of real functions, *Theoret. Comput. Sci.*, **20** (1982) 323–352.
- [85] G. Kreisel, Analysis of the Cantor-Bendixson theorem by means of the analytic hierarchy, *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, **7** (1959) 621–626.
- [86] L. Kronecker, Grundzüge einer arithmetischen Theorie der algebraischen Grössen, *J. Reine Angew. Math.*, **92** (1882) 1–122.
- [87] B. Kušner, Riemann integration in constructive analysis (Russian), *Dokl. Akad. Nauk SSSR*, **156** (1964) 255–257; [translated in: *Soviet Math. – Dokl.*, **5** (1964) 628–630].
- [88] D. Lacombe, Classes récursivement fermées et fonctions majorantes, *C. R. Acad. Sci. Paris*, **240** (1955) 716–718.
- [89] R. Ladner, On the structure of polynomial time reducibility, *J. Assoc. Comput. Mach.*, **22** (1975) 155–171.
- [90] C. R. Lin, The effective content of Ulm’s Theorem, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981) 147–160.
- [91] C. McCarty, Realizability and recursive set theory, *Ann. Pure Appl. Logic*, **32** (1986) 153–183.
- [92] A. I. Mal’tsev, Constructive algebras I (Russian), *Uspekhi Mat. Nauk*, **16** (1961) 3–60; [translated in: *Constructive algebras I*, *Russian Math. Surveys*, **16**:3 (1961) 77–129; also in: *The Metamathematics of Algebraic Systems*, *Collected Papers: 1936–1967*, translated and edited by B. F. Wells III, *Stud. Logic Found. Math.*, **66** (1971), Ch. 18, 148–200].

- [93] A. I. Mal'tsev, Algorithms and Recursive Functions (Russian), (Izdat. Nauka, Moscow, 1965); [translated: by L. F. Boron, L. E. Sanchis, J. Stillwell and K. Iseki, (Wolters-Noordhoff Publishing, Groningen, 1970)].
- [94] A. I. Mal'tsev, Algebraic Systems (Russian), (posth. edn.), V. D. Smirnov and M. Tačlin, (eds.), (Izdat. Nauka, Moscow, 1970); [translated: by B. D. Seckler and A. P. Doohovskoy, Grundlehren Math. Wiss., **192** (1973).]
- [95] A. B. Manaster and J. G. Rosenstein, Effective matchmaking (recursion theoretical aspects of a theorem of Philip Hall, Proc. London Math. Soc., Ser. 3, **25** (1972) 615–654.
- [96] A. B. Manaster and J. G. Rosenstein, Effective matchmaking and k -chromatic graphs, Proc. Amer. Math. Soc., **39** (1973) 371–378.
- [97] A. A. Markov, On the representation of recursive functions (Russian), Dokl. Akad. Nauk SSSR (N.S.), **58** (1947) 1891–1892.
- [98] A. A. Markov, The Theory of Algorithms (Russian), Izdat. Akad. Nauk SSSR, Moscow, Trudy Mat. Inst. Steklov., **42** (1954); [translated: by J. J. Schorr-Kon and PST staff, publ. for NSF, Washington, DC, and the Dept. Commerce by (Israel Prog. for Scientific Translations, Jerusalem, 1961).]
- [99] W. Markwald, Zur Theorie der konstruktiven Wohlornungen, Math. Ann., **127** (1954) 135–149.
- [100] Yu. V. Matijasevič, Enumerable sets are diophantine (Russian), Dokl. Akad. Nauk SSSR, **191** (1970) 279–282; [translated in: Soviet Math. – Dokl., **11** (1970) 354–357].
- [101] S. Mazur, Computable Analysis, (based on lecture notes academic year 1949/50, Inst. Math., Polish Acad. Sci., Warsaw), A. Grzegorzcyk and H. Rasiowa, (eds.), Rozprawy Matematyczne, Vol. 33, (Państwowe Wydawn. Naukowe, Warszawa, 1963).
- [102] G. Metakides and A. Nerode, Recursion theory and algebra, in: Algebra and Logic, (Proc. 14th. Summer Res. Inst. Austral. Math. Soc., Monash Univ., Clayton, Vic., Australia, Jan. 6 – Feb. 16, 1974), J. N. Crossley, (ed.), Lecture Notes in Math., **450** (1975) 209–219.
- [103] G. Metakides and A. Nerode, Recursively enumerable vector spaces, Ann. Math. Logic, **11** (1977) 147–171.

- [104] G. Metakides and A. Nerode, Effective content of field theory, *Ann. Math. Logic*, **17** (1979) 289–320.
- [105] G. Metakides and A. Nerode, Recursion theory on fields and abstract dependence, *J. Algebra*, **65** (1980) 36–59.
- [106] T. S. Millar, The Theory of Recursively Presented Models, Ph.D. Thesis, Cornell Univ., Ithaca, NY, (1976).
- [107] T. S. Millar, Foundations of recursive model theory, *Ann. Math. Logic*, **13** (1978) 305–320.
- [108] A. A. Mučnik, On the unsolvability of the problem of reducibility in the theory of algorithms (Russian), *Dokl. Akad. Nauk SSSR*, **108** (1956) 194–197.
- [109] J. Myhill, Creative sets, *Z. Math. Logik Grundlag. Math.*, **1** (1955) 97–108.
- [110] A. Nerode, Extensions to isols, *Ann. of Math.*, **73** (1961) 362–403.
- [111] A. Nerode and J. B. Remmel, Recursion theory on matroids. in: Patras Logic Symposion, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) G. Metakides (ed.), *Stud. Logic Found. Math.*, **109** (1982) 41–65.
- [112] A. Nerode and J. B. Remmel, A survey of lattices of r.e. substructures, in: Recursion Theory, (Proc. AMS-ASL Summer Inst., Ithaca, NY, June 28 – July 16, 1982), A. Nerode and R. A. Shore, (eds.), *Proc. Sympos. Pure Math.*, **42** (1985) 323–375.
- [113] A. Nerode and J. B. Remmel, Complexity theoretic algebra I: Vector spaces over finite fields, in: Structure in Complexity Theory, (Proc. 2nd. Annual IEEE Conf. Struct. Complexity Theory, Cornell Univ., Ithaca NY, June 16–19, 1987), (IEEE Comput. Soc. Press, Washington, DC, 1987), 218–239.
- [114] A. Nerode and J. B. Remmel, Complexity-theoretic algebra II: Boolean algebras, (Special issue: 3rd. Asian Conf. Math. Logic, Beijing, Oct. 26–30, 1987), D. P. Yang, (ed.), *Ann. Pure Appl. Logic*, **44** (1989) 71–99.
- [115] A. Nerode and J. B. Remmel, Complexity-theoretic algebra: vector space bases, in: Feasible Mathematics, (Proc. Workshop, June 16–18, 1989, Cornell Univ. Ithaca, NY), S. Buss and P. J. Scott (eds.), *Progr. Comput. Sci. Appl. Logic*, **9** (1990) 293–319.

- [116] A. Nerode and J. B. Remmel, Polynomial time equivalence types, in: *Logic and Computation*, (Proc. Workshop, CMU, Pittsburg, June 30 – July 2, 1987), W. Sieg, (ed.), *Contemp. Math.*, **106** (1990) 221–249.
- [117] P. S. Novikov, On the algorithmic unsolvability of the word problem in group theory (Russian), *Trudy Mat. Inst. Steklov.*, **44** (1955).
- [118] V. P. Orevkov, Certain types of continuity of constructive operators (Russian), *Trudy Mat. Inst. Steklov.*, **93** (1967) 164–186.
- [119] R. Peter, *Rekursive Funktionen*, (Akademischer Verlag, Budapest, 1951).
- [120] E. L. Post, Formal reductions of the general combinatorial decision problem, *Amer. J. Math.*, **65** (1943) 197–215.
- [121] E. L. Post, Recursively enumerable sets of positive integers and their decision problems, *Bull. Amer. Math. Soc. (N.S.)*, **50** (1944) 284–316.
- [122] M. B. Pour-El and J. I. Richards, Computability in Analysis and Physics, *Perspect. in Math. Logic*, (1989).
- [123] M. Presburger, Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt, *Comptes-Rendus du I Congres des Mathématiciens des Pays Slaves*, Warszawa, (1929) 92–101, 393.
- [124] M. O. Rabin, Recursive unsolvability of group theoretic problems, *Ann. of Math.*, **67** (1958) 172–194.
- [125] M. O. Rabin, Computable algebra, general theory and theory of computable fields, *Trans. Amer. Math. Soc.*, **95** (1960) 341–360.
- [126] J. B. Remmel, Combinatorial functors on co-r.e. structures, *Ann. Math. Logic*, **10** (1976) 261–287.
- [127] J. B. Remmel, Co-hypersimple structures, *J. Symbolic Logic*, **41** (1976) 611–625.
- [128] J. B. Remmel, Recursively enumerable Boolean algebras, *Ann. Math. Logic*, **15** (1978) 75–107.
- [129] J. B. Remmel, Recursion theory on algebraic structures with independent sets, *Ann. Math. Logic*, **18** (1980) 153–191.

- [130] J. B. Remmel, Recursive Boolean algebras, in: *Handbook of Boolean Algebras*, Vol. 3, J. D. Monk, R. Bonnet and S. Koppelberg, (eds.), (North-Holland, Amsterdam, NY, 1989), Ch. 25, 1097–1165.
- [131] A. Retzlaff, Direct summands of recursively enumerable vector spaces, *Z. Math. Logik Grundlag. Math.*, **25** (1979) 363–372.
- [132] H. G. Rice, Recursive and recursively enumerable orders, *Trans. Amer. Math. Soc.*, **83** (1956) 277–300.
- [133] J. Robinson, Definability and decision problems in arithmetic, *J. Symbolic Logic*, **14** (1949) 98–114.
- [134] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, (1st. edn., McGraw-Hill, New York-Toronto, Ont.-London, 1967; 2nd. edn., MIT Press, Cambridge, Mass., London, 1987).
- [135] J. G. Rosenstein, Recursive linear orderings, in: *Orders: Description and Roles*, *Ann. Discrete Math.*, **23** (1984) 465–475.
- [136] G. E. Sacks, The recursively enumerable degrees are dense, *Ann. of Math.*, **80** (1964) 300–312.
- [137] N. A. Šanin, Some problems of mathematical analysis in the light of constructive logic, *Z. Math. Logik Grundlag. Math.*, **2** (1956) 27–36.
- [138] W. Savitch, Relationships between nondeterministic and deterministic tape complexities, *J. Comput. System Sci.*, **4** (1970) 177–192.
- [139] J. R. Shoenfield, Degrees of formal systems, *J. Symbolic Logic*, **23** (1958) 389–392.
- [140] J. R. Shoenfield, Degrees of models, *J. Symbolic Logic*, **25** (1960) 233–237.
- [141] R. A. Shore, Controlling the dependence degree of a recursively enumerable vector space, *J. Symbolic Logic*, **43** (1978) 13–22.
- [142] R. I. Soare, Recursion theory and Dedekind cuts, *Trans. Amer. Math. Soc.*, **140** (1969) 271–294.
- [143] E. Specker, Ramsey's Theorem does not hold in recursive set theory, in: *Logic Colloq. '69*, (1971) 439–442.
- [144] C. Spector, Recursive well-orderings, *J. Symbolic Logic*, **20** (1955) 151–163.

- [145] L. J. Stockmeyer, The polynomial-time hierarchy, *Theoret. Comput. Sci.*, **3** (1976) 1–22 (1977).
- [146] A. Tarski, Undecidability of group theory, *J. Symbolic Logic*, **14** (1949) 76–77.
- [147] A. Tarski, A Decision method for elementary algebra and geometry, 2nd. edn., revised, (Univ. California Press, Berkeley and Los Angeles, CA, 1951).
- [148] A. Tarski, *Logic, Semantics, Metamathematics*, Papers from 1923 to 1938, (Oxford Univ. Press, 1956).
- [149] S. Tennenbaum, Non-archimedean models for arithmetic, *Notices Amer. Math. Soc.*, **6** (1959) 270.
- [150] J. Thurber, *Degrees of Boolean Algebras*, Ph.D. Thesis, Univ. Notre Dame, Notre Dame, IN, (1994).
- [151] J. J. Thurber, Recursive and r.e. quotient Boolean algebras, *Arch. Math. Logic*, **33** (1994) 121–129.
- [152] A. S. Troelstra and D. van Dalen, *Constructivism in Mathematics*, Vol. 1, *Stud. Logic Found. Math.*, **121** (1988).
- [153] A. S. Troelstra and D. van Dalen, *Constructivism in Mathematics*, Vol. 2, *Stud. Logic Found. Math.*, **123** (1988).
- [154] A. M. Turing, On computable numbers, with an application to the Entscheidungsproblem, *Proc. London Math. Soc.*, Ser. 2, **42** (1936) 230–265; corr. *ibid.*, **43** (1937) 544–546.
- [155] B. L. van der Waerden, Eine Bemerkung über die Unzerlegbarkeit von Polynomen, *Math. Ann.*, **102** (1930) 738–739.
- [156] R. L. Vaught, Sentences true in all constructive models, *J. Symbolic Logic*, **25** (1960) 39–58.
- [157] B. L. van der Waerden, *Moderne Algebra*, Parts I and II, (G. E. Stechert and Co., New York, 1943); [translated as: *Modern Algebra*, Vols. I and II, (translated from the 2nd. rev. German edition by F. Blum, with revisions and additions by the author), Frederick Ungar Publishing Co., NY, (1949)].

- [158] B. L. van der Waerden, *Algebra*, Vols. I and II, (7th. edn. (1966) Vol. I, 5th. edn. (1967) Vol II), *Heidelberger Taschenbücher* **12** and **23**, Springer-Verlag, Berlin and NY; [Vol. I translated by F. Blum and J. R. Schulenberg, Vol. II translated by J. R. Schulenberg, (Frederick Ungar Publishing Co., NY, 1970), reprinted: Springer-Verlag, NY, 1991].

PART TWO

RECURSIVE ALGEBRA, ANALYSIS

and

COMBINATORICS

This Page Intentionally Left Blank

Chapter 13

Π_1^0 Classes in Mathematics

D. Cenzer and J. B. Remmel*

Contents

- Introduction
- 1. Preliminaries
- 2. Π_1^0 classes and their members
- 3. Logical theories
- 4. Non-monotone logics
- 5. Recursive algebra
- 6. Recursive graphs
- 7. Recursive partial orderings
- 8. Recursive orderings and ordered structures
- 9. Recursive analysis
- 10. Gale-Stewart games and the Rado selection principle
- 11. Feasible Π_1^0 classes
- 12. Feasible models
- References

Introduction

Recursive mathematics started in the 1950's with the work of Specker [145] and Lacombe [94] on recursive analysis and the work of Frölich and Shepherdson [55] and Rabin [124] on effective field theory. In the 1970's, Metakides and Nerode [110] introduced the priority method into the study of recursive algebra and it has been an active area of research ever since. A central goal

*The work of the second author was partly supported by Dept. of Commerce Agreement 70-NANB5H1164 and NSF grant DMS-9306427.

of recursive mathematics to study the effective content of constructions and proofs in various branches of mathematics including combinatorics, analysis, topology, algebra and model theory. In general, one examines a classical result which provides a solution for a problem associated with some mathematical structure and studies whether an effective (or recursive) solution can be obtained if the structure itself is effective. For example, one might consider the classical result that every independent subset of a vector space V can be extended to a basis for V . The natural analogue of this result that one would study in recursive algebra is whether every recursive independent set of a recursive vector space V can be extended to a recursive basis for V . Since all the proofs of the fact that every independent subset of vector space V can be extended to a basis of V use some form of the axiom of choice, e.g., Zorn's lemma, when V is infinite dimensional, and it is well known that the axiom of choice is nonconstructive, one might naturally expect that it is not the case that every recursive independent subset of an infinite dimensional recursive vector space V can be extended to a recursive basis for V . Indeed, this was demonstrated by Metakides and Nerode [111]. Recursive algebra has been studied by a large number of authors; see the survey article by Nerode and Remmel [116]. For another example, consider the classical four color theorem of Appel and Haken [2, 3, 4] which shows that any planar graph can be colored with four colors. From the recursion theoretic point of view, we would like to know whether a recursive planar graph has a recursive 4-coloring and whether there is a general algorithm for obtaining this coloring. More generally one might ask whether every k -colorable recursive graph G has a recursive k -coloring and, if it is not the case that G has a recursive k -coloring, what can we say about the Turing degrees of the k -colorings of G . Recursive combinatorics, including the analysis of the complexity of algorithms for solving such problems as the coloring problem and the marriage problem, was developed by Kierstead [77, 78, 80], Bean [7, 8], Remmel [128, 129], Schmerl [134, 135] and others. See the survey article by Gasarch [57] in this volume.

Given the influence of complexity theory from Computer Science, it is natural to study various classes of feasible structures such as polynomial time structures, i.e., structures whose universe, relations, and operations are all polynomial time computable. Recently, a version of feasible mathematics has begun to be developed along the lines of recursive mathematics. The primary notion of feasibility has been that of polynomial-time computability. For example, there has been a considerable development of complexity theoretic

analysis starting with the work of Friedman and Ko [53], see Ko's book [88]. Similarly Cenzer and Remmel [21, 23, 24, 22] have developed a notion of complexity theoretic model theory and Nerode and Remmel [117, 118, 119, 120] have developed complexity theoretic algebra; see the survey article by Cenzer and Remmel [25]. Here one looks for similarities and differences between recursive and feasible structures and problems. For instance, the following questions have been studied by Cenzer and Remmel.

- (A) Is every recursive structure isomorphic to a polynomial structure?
- (B) Is every recursive structure isomorphic to a polynomial time structure over a standard universe such as tally representation of the natural numbers $Tal(\omega)$ or the binary representation of the natural numbers $Bin(\omega)$.
- (C) Is every recursive structure recursively isomorphic to a polynomial structure?
- (D) Is every recursive structure recursively isomorphic to a polynomial time structure over a standard universe such as tally representation of the natural numbers $Tal(\omega)$ or the binary representation of the natural numbers $Bin(\omega)$.

Cenzer and Remmel showed that every recursive relational structure is recursively isomorphic to a polynomial time relational structure and constructed recursive structures with function symbols which are not isomorphic to any polynomial time structure. Grigorieff [61] showed that every recursive linear ordering is isomorphic to a polynomial time structure with universe $Bin(\omega)$ while Cenzer and Remmel showed that there is a recursive linear ordering of order type $\omega + \omega^*$ which is not recursively isomorphic to any polynomial time linear ordering whose universe is $Bin(\omega)$. Other areas of interest in feasible mathematics include the question of whether a feasible problem will have a feasible solution and the classification of feasibly categorical structures.

The study of Π_1^0 classes arises naturally in the study of recursive mathematics. An example of the role played by Π_1^0 classes in the study of recursive mathematics is the following. In many combinatorial and algebraic problems, the family of solutions to the given problem (such as the family of 4-colorings of a given planar graph) can be viewed as a closed set under some natural topology. Now a Π_1^0 class is simply an effectively closed set. Thus for many

recursive combinatorial and algebraic problems, the set of solutions can be viewed as a Π_1^0 class. Moreover, if the classical set of solutions is compact, then the effective version will often be a *bounded* or *recursively bounded* Π_1^0 class. Thus one is naturally led to the study of three types of Π_1^0 classes, namely arbitrary Π_1^0 classes, bounded Π_1^0 classes, and recursively bounded Π_1^0 classes, and their members.

Let $\omega = \{0, 1, 2, \dots\}$ denote the set of natural numbers. A Π_1^0 class P is simply an effectively closed subset of the Baire space ω^ω of infinite sequences of natural numbers. For our purposes, we will define a Π_1^0 class to be the set $[T]$ of all infinite paths through a recursive tree T . Here a recursive tree is a recursive subset T of $\omega^{<\omega}$, the set of all finite sequences from the natural numbers ω , which contains the empty string and is closed under initial segments. If T is finitely branching, the corresponding Π_1^0 class is said to be *bounded*. A recursive tree T is said to be *highly recursive* and the corresponding Π_1^0 class $[T]$ is said to be *recursively bounded* (r.b.) if T is finite branching and there is a partial recursive function $f : T \rightarrow \omega$ such that, for each node σ of T , $f(\sigma)$ is the number of successors of σ . A tree T is said to be *highly recursive in $\mathbf{O}'$* if T is recursive in $\mathbf{O}'$, T is finite branching and there is a $\mathbf{O}'$ -recursive function $f : T \rightarrow \omega$ such that, for each node σ of T , $f(\sigma)$ is the number of successors of σ . In this paper, we will consider primarily bounded and recursively bounded Π_1^0 classes. We will also discuss the connections between arbitrary Π_1^0 classes, bounded Π_1^0 classes, and recursively bounded Π_1^0 classes. For example, clearly every finitely branching recursive tree is highly recursive in $\mathbf{O}'$. Moreover a result of Jockusch, Lewis, and Remmel shows that for every highly recursive in $\mathbf{O}'$ tree T , there is a finitely branching recursive tree S such that there is an effective one-to-one correspondence between the elements of $[T]$ and the elements of $[S]$. Thus many results about the degrees of elements of bounded Π_1^0 classes can be derived from results about the degrees of recursively bounded Π_1^0 classes by relativizing those results to a $\mathbf{O}'$ oracle.

Π_1^0 classes occur in many areas of recursive mathematics. We have not attempted to provide an exhaustive survey of all applications of Π_1^0 classes to mathematics. We will, however, present applications in a wide variety of fields including logic, nonmonotonic logic, algebra, combinatorics, orderings and game theory. In each case, we will show that the set of solutions to a given recursive instance of the problem may be represented as an arbitrary (bounded, or recursively bounded) Π_1^0 class. For example, Bean [7] observed that the set of k -colorings of a recursive graph G is a recursively bounded

Π_1^0 class. Here $G = (V, E)$ is a *recursive graph* if the set V of vertices is a recursive subset of ω and the set E of edges is a recursive subset of $[V]^2$, the set of unordered pairs of vertices. Generally one would like a converse of this type of result. That is, given an arbitrary (bounded, or recursively bounded) Π_1^0 class, one would like to show that there is a recursive instance of the problem such that there is a one-to-one degree preserving correspondence between the solutions to the problem and the elements of the Π_1^0 class. For example, Remmel [129] showed that any r.b. Π_1^0 class may be represented, up to a permutation of the colors, by the set of k -colorings of a recursive graph for $k \geq 3$. This implies that in some sense, the problem of finding a k -coloring of a recursive k -colorable graph and the problem of finding an element of an r.b. Π_1^0 class are equivalent. We will define precisely the ways in which a Π_1^0 class may be “represented” by the set of solutions to a given recursive problem below. Much of the paper will be devoted to cataloging and proving such representation results. However in some applications, these questions are as yet unanswered. In each case, the analysis of Π_1^0 classes provides information about the complexity of the solutions of the various recursive mathematical problems.

This paper is intended mainly to be a survey of recent research in the study of Π_1^0 classes and their application to recursive mathematics. However new results will be given on the representation of Π_1^0 classes as the solution sets to various mathematical problems. There will also be some new results on feasible problems and solutions.

The primary question that we will examine for any given Π_1^0 class P is the determination of the possible degrees of the elements of P . This problem has been studied by many recursion theorists, going back to the Kleene basis theorem, which showed that every Π_1^0 class contains a member which is recursive in some Σ_1^1 set, and the Kreisel-Shoenfield basis theorem [136], which showed that every r.b. Π_1^0 class contains a member of degree $\leq \mathbf{0}'$. Two important early papers in this area are [75, 74] by Jockusch and Soare. They show, among other things, that there is a Π_1^0 class with no recursive members and such that any two members have mutually incomparable Turing degrees. An application of this result to recursive combinatorics naturally occurs when this result is combined with Remmel’s results, that every recursively bounded Π_1^0 class can be represented as the 3-colorings of a recursive graph, to show that there is a recursive 3-colorable graph with no recursive 3-colorings and such that any two 3-colorings which do not differ merely by a permutation of the colors have incomparable Turing degrees.

As part of our goal to study the relation between Π_1^0 classes and their members, we will study the relationship between elements of a Π_1^0 class and the Cantor-Bendixson rank of its elements. Kreisel first noticed in [92] that the degree of a member x of a Π_1^0 class is related to the Cantor-Bendixson rank of x in P , when he showed that every member of a countable Π_1^0 class is hyperarithmetical. This relationship has been developed in detail in recent work on countable Π_1^0 classes and Cantor-Bendixson rank by Cenzer, Clote, Smith, Soare and Wainer [16, 28] and more recently by Cholak and Downey [30] as well as work by Cenzer, Downey, Jockusch and Shore on countable thin Π_1^0 classes [17].

A Π_1^0 class P is said to be *thin* if every Π_1^0 subclass Q of P is the intersection of P with some clopen set U . P is said to be *minimal* if every Π_1^0 subclass of P is either finite or cofinite in P . One of the elementary properties of Π_1^0 classes is that any isolated member of a Π_1^0 class must be recursive. This implies that all of the elements of a finite Π_1^0 class are recursive. Now a thin Π_1^0 class has the converse property that any recursive member is isolated. Countably infinite thin Π_1^0 classes were constructed in [17]. This construction can be applied to the coloring problem to demonstrate the existence of a recursive 3-colorable graph which has infinitely many recursive 3-colorings, each of which is uniquely determined by its restriction to a finite subgraph.

Once we have set up the framework of Π_1^0 classes, we will develop the connection between Π_1^0 classes and recursive mathematical problems as outlined above. This connection leads to a series of natural questions about whether the set of solutions to a given problem can be represented as a Π_1^0 class and conversely, whether a given problem can represent an arbitrary recursively bounded Π_1^0 class. This is the main concern in the second part of the paper.

One primary emphasis will be on problems in recursive combinatorics. The reason that r.b. Π_1^0 classes arise so naturally in the study of recursive combinatorics is due to the fact that many combinatorial theorems about finite graphs and partially ordered sets (posets) can be extended to countably infinite graphs and posets by applying König's lemma, i.e., the fact that every infinite finitely branching tree has an infinite path through it. The basic idea of such extensions, which will be made explicit in Section 6, is that for each instance of a combinatorial problem P , we can construct a finitely branching tree T so that any solution to the combinatorial problem corresponds to an infinite path through T . Then we use the fact that the theorem holds for all finite instances to prove that the tree T is infinite. For example, we may use such an idea to prove that any countably infinite graph G is k -colorable if

and only if every finite subgraph of G is k -colorable and hence deduce that every countable planar graph is 4-colorable from the fact that every finite planar graph is 4-colorable. Here the combinatorial problem is to find the 4-colorings of the graph. In such a case one starts with a graph G and builds T so that the infinite paths through T correspond to the k -colorings of G . In many cases, if the combinatorial problem P is effectively or recursively presented, then the corresponding tree T is highly recursive and hence the set of solutions to the combinatorial problem will be an r.b. Π_1^0 class. For example, Bean [7] observed this to be the case for the problem of k -coloring a recursive graph G . We will also consider the Hamiltonian circuit problem, the vertex partition problem of Ulam, the marriage problem of Philip and Marshall Hall [63] and other matching problems associated with recursive graphs. Three problems associated with posets to be considered are the dual problems of covering a poset with chains (Dilworth [34]) or with antichains and the dimension problem (Dushnik and Miller ([46]) of expressing a poset as the intersection of linear orderings.

As mentioned above, Jockusch and Soare [75, 74] gave a detailed analysis of the possible Turing degrees of elements of r.b. Π_1^0 classes. The above discussion shows that their work can be used to provide an analysis of the complexity of the solutions to certain infinite recursive combinatorial problems. The key fact that one must establish, to be able to transfer known results on the degrees of elements of Π_1^0 classes to results about the degrees of solutions to a given type of recursive combinatorial problem, is that every r.b. Π_1^0 class can be represented as the set of solutions to a specific instance of the combinatorial problem. More specifically, we need to show that for each highly recursive finitely branching tree T , there is an effective one-to-one correspondence between infinite paths through T and solutions to some recursively presented instance P of the combinatorial problem. Of course, in general, this is too much to ask, since for example, if we are considering k -colorings of a graph or the coverings of poset by k chains, we always have multiple solutions due to the fact that permuting the colors or names of the chains leads from one solution to other solutions. Thus, a more reasonable problem is to ask whether, for each highly recursive finitely branching tree T , there is, say, a recursive graph $\mathcal{G}$ or a recursive poset $\mathcal{P}$, such that, up to a permutation of the labels of the colors or chains, there is a recursive one-to-one correspondence between the k -colorings of $\mathcal{G}$ or the coverings of $\mathcal{P}$ by k chains, and the infinite paths through T . Without being completely formal at this point, we will refer to the ability of specific recursively presented instances

of a combinatorial problem P to represent an arbitrary r.b. Π_1^0 class in the sense above as P *strongly representing* every r.b. Π_1^0 class.

Now if a combinatorial problem P fails to strongly represent every r.b. Π_1^0 class, it still may be the case that, for each highly recursive finitely branching tree T , there is a specific recursively presented instance of P such that the set of Turing degrees of solutions to the recursively presented instance of P equals the set of Turing degrees of infinite paths through T . Under these latter circumstances, we shall say that P *degree represents* every r.b. Π_1^0 class. Finally we shall say that P *weakly represents* every r.b. Π_1^0 class if for any recursive tree S , there is a recursive instance P_S of P such that there are recursive functionals φ and ψ such that for every solution s to P_S , $\varphi(s) \in [S]$ and $\varphi(s) \equiv_T s$ and for every $x \in [S]$, $\psi(x)$ is a solution to P_S and $x \equiv_T \psi(x)$. It is rather trivial to find examples of r.b. Π_1^0 classes which can weakly but not strongly be represented by a combinatorial problem. For example, every recursive connected 2-colorable graph has a unique 2-coloring up to a permutation of the colors and hence can weakly but not strongly represent a highly recursive tree T which has only a countably infinite set of infinite paths through it, all of which are recursive. Furthermore, the problem of 2-coloring recursive connected graphs cannot degree represent or even weakly represent an r.b. Π_1^0 class with only nonrecursive elements for, as Bean observed in [7], a recursive connected graph is 2-colorable if and only if it is recursively 2-colorable. A similar situation occurs for a certain natural matching problem discussed in Section 6. Jockusch was the first to raise such representation questions in [69] by conjecturing, in our language, that the marriage problem corresponding to the Philip Hall Theorem can strongly represent every r.b. Π_1^0 class. This question still remains open. We note the notions of a combinatorial problem strongly representing, degree representing, or weakly representing any bounded Π_1^0 class or any arbitrary Π_1^0 class are defined similarly.

Of course recursion theory itself is a part of mathematics and Π_1^0 classes occur in several places in recursion theory. The study of r.e. sets and degrees has been one of the primary topics in recursion theory in recent years. (See for example, Soare's book [143].) Now for any r.e. set A , the family $\mathcal{S}(A)$ of supersets of A forms a Π_1^0 class. Thus the incredibly rich structure of r.e. sets can be embedded in the structure of Π_1^0 classes. For a co-r.e. set C , the family of subsets of C will also be a Π_1^0 class. These examples of Π_1^0 classes are studied by Cenzer, Downey et. al. in [17], where the notion of a *minimal*

Π_1^0 class is proposed and studied, in part as an analogy to the notion of a maximal r.e. set. A related problem in recursion theory which leads to a Π_1^0 class is the problem of separating a given pair of disjoint r.e. sets. Shoenfield observed in [136] that for any pair (A, B) of disjoint r.e. sets, the family of sets C which contain A and are disjoint from B , i.e., the family of separating sets for the pair (A, B) is always a Π_1^0 class. Not every r.e. Π_1^0 class can be represented as a class of separating sets. However, some of the existence results for Π_1^0 classes with specified properties have been modified to produce classes of separating sets. For example, Jockusch and Soare show in [75] that there are disjoint coinfinite r.e. sets A, B such that any two separating sets are Turing incomparable unless they differ by a finite set. Thus a problem which can represent every class of separating sets must have an instance for which the solutions (modulo finite difference) all have different Turing degree. We will show that several of the combinatorial problems listed above can strongly represent the Π_1^0 class of separating sets for any pair of r.e. sets.

Another area of particular interest is the connection between logical theories, recursively enumerable Boolean algebras, commutative rings with unity and Π_1^0 classes. The problem associated with a logical theory is to find a complete consistent extension of that theory. This has been the problem of the most interest in connection with Π_1^0 classes. For example, the classical undecidability theorem of Turing and Church can be viewed as showing that the Π_1^0 class of complete consistent extensions of Peano Arithmetic has no recursive member. Shoenfield [137] showed in 1960 that the set of complete consistent extensions of a decidable first-order theory is always a Π_1^0 class. Ehrenfeucht [47] showed in 1961 that, conversely, every Π_1^0 class is represented by the set of complete consistent extensions of some decidable first-order theory.

In nonmonotonic logic, Marek, Nerode, and Remmel [104] showed that the set of stable models of a recursive general logic program or the set of extensions of a recursive default theory can each represent an arbitrary Π_1^0 class. Moreover there are natural subclasses of recursive logic programs or recursive default theories which can represent bounded or recursively bounded Π_1^0 classes. Thus nonmonotonic logic provides a particular rich application of the theory of Π_1^0 classes.

Now the set of sentences in any first-order language may be viewed as a recursive Boolean algebra modulo truth-table equivalence. That is, the Lindenbaum algebra of a first-order theory T is defined to be the set of sentences of the language modulo provable equivalence in T . If the theory T

is axiomatizable, so that the set of consequences is recursively enumerable, then this Boolean algebra will be an r.e. Boolean algebra. (Here we say that a Boolean algebra is *recursively enumerable* (r.e.) if it is the quotient of a recursive Boolean algebra modulo an r.e. filter.) In this Boolean algebra, a proper filter is a consistent theory, a recursive filter is a decidable theory, a recursively enumerable filter is an axiomatizable theory, and an ultrafilter is a complete, consistent theory. Hence the problem which we will consider for a Boolean algebra $\mathcal{B}$ is to find an ultrafilter for $\mathcal{B}$. Note that the Undecidability Theorem proves the existence of an r.e. Boolean algebra with no recursive ultrafilters, or, equivalently, an r.e. filter with no extension to a recursive ultrafilter. The result of Ehrenfeucht shows that any r.b. Π_1^0 class can be represented as the set of ultrafilters of an r.e. Boolean algebra. We will give a simpler, direct proof of this result. The standard results of Jockusch and Soare can then be applied to obtain, for example, an r.e. Boolean algebra $\mathcal{B}$ such that any two distinct ultrafilters of $\mathcal{B}$ are Turing incomparable. If the Π_1^0 class is thin, then the Boolean algebra will have the property that any r.e. filter is principal. On the other hand, it is easy to see that the class of ultrafilters of any r.e. Boolean algebra $\mathcal{B}$ can always be represented by a Π_1^0 class—the Stone space of $\mathcal{B}$. It now follows, for example, that any r.e. Boolean algebra $\mathcal{B}$ has an ultrafilter which is recursive in $0'$ and that if $\mathcal{B}$ has only countably many ultrafilters, then $\mathcal{B}$ has a recursive ultrafilter. In addition, there is a version of the correspondence between r.b. Π_1^0 classes and r.e. Boolean algebras which associates Π_1^0 classes $P = [T]$, where T is a recursive tree with no dead ends, with recursive Boolean algebras. That is, for any recursive Boolean algebra, the family of ultrafilters of $\mathcal{B}$ can be represented by the Π_1^0 class $[T]$ of infinite paths through some highly recursive tree T with no dead ends and, conversely, for any highly recursive tree T with no dead ends, there is a recursive Boolean algebra $\mathcal{B}$ such that the family of ultrafilters of $\mathcal{B}$ is represented by $[T]$. A simple consequence of this correspondence is the fact that every recursive Boolean algebra has a recursive ultrafilter.

Now any Boolean algebra also corresponds to a Boolean ring, and the results above can be phrased in terms of ideals and in particular, prime and maximal ideals. We consider the problem of finding ideals for commutative rings, studied by Friedman, Simpson and Smith [54], along with the similar problem of finding subgroups of Abelian groups.

The set of possible orderings of recursive ordered algebraic structures also form a Π_1^0 class. For example, Metakides and Nerode [112] proved that an arbitrary r.b. Π_1^0 class can be represented by the orderings of a formally

real recursive field R . We consider also the related problem of orderings of Abelian groups, studied by Downey and Kurtz [42].

There are several other problems associated with linear orderings which lead to non-recursively-bounded Π_1^0 classes. These include the problem of finding suborderings of order type ω , the problem of finding ω -successivities and the problem of finding self-embeddings.

Another interesting problem is that of finding a winning strategy for a closed game. The classical theorem of Gale and Stewart [56] states that any open/closed two-player game of perfect information is determined. Cenzer and Remmel [24] gave a representation for the family of winning strategies by a Π_1^0 class and showed that any r.b. Π_1^0 class can be strongly represented by such a family of winning strategies.

There are many important problems in recursive analysis associated with a continuous real function F which naturally lead to Π_1^0 classes. For example, we will consider the problems of finding a zero of F , finding a maximum value for F and finding a fixed point of F . Recursive analysis has been studied ever since Lacombe [94], who showed that there are recursive real functions which have zeroes but have no recursive zeroes. We will improve this result by showing that any r.b. Π_1^0 class may be represented as the set of zeroes of some recursive real function. We will also consider the problems from dynamical systems of finding attracting points and their basins of attraction. For example, we will show that the well-known Mandelbrot set of complex numbers z such that the iterated image $\{z, f(z), f(f(z)), \dots\}$ of z , under the recursive complex function $f(z) = z^2 + c$, is bounded, can be represented by an r.b. Π_1^0 class.

The final topic of the paper is feasible problems and feasible Π_1^0 classes. In particular, we examine similarities and the differences between the solution sets of feasible problems and the solution sets of non-feasible problems.

This paper is organized as follows. Section one contains preliminaries and notation. Section two contains the survey of results on Π_1^0 classes and their members. Sections three through ten contain results on various recursive problems in mathematics where, for the most part, the solution sets are shown to be represented by r.b. Π_1^0 classes and, in some cases, to represent all possible r.b. Π_1^0 classes. (Certain problems are represented by bounded or unbounded Π_1^0 classes.) The results of Section two are then applied to derive corollaries such as the existence of recursive instances of each such problem with no recursive solutions. Section eleven contains the results from feasible model theory which will be needed to analyze feasible versions of the

mathematical problems and Section twelve has results on feasible versions of the many of the mathematical problems considered in the previous sections. For most such problems, we will show that for any given recursive instance R of the problem, there is a polynomial-time instance P of the problem and a one-to-one degree preserving correspondence between the solutions of P and the solutions of R . We then apply the results of Section eleven to derive corollaries such as the existence of a feasible instance of such problems with no recursive solutions.

1 Preliminaries

In this section, we shall establish our notation and the basic definitions from recursion theory and complexity theory that will be needed for the subsequent sections.

Let $\omega = \{0, 1, \dots\}$. Let Σ be a (usually finite) alphabet. Then Σ^* denotes the set of finite strings of letters from Σ , and Σ^ω denotes the set of infinite sequences. In particular, each natural number n may be represented in unary form by the string $tal(n) = 1^n$ if $n > 0$ and $tal(n) = 0$ if $n = 0$, and in (reverse) binary form by the string $bin(n) = i_0 \cdots i_k$, where $n = i_0 + i_1 \cdot 2 + \cdots + i_k \cdot 2^k$.

We let $Tal(\omega) = \{tal(n) : n \in \omega\}$ and $Bin(\omega) = \{bin(n) : n \in \omega\}$. Both sets are included in $\{0, 1\}^*$. The tally and binary representation of the natural numbers will be essential for our study of feasible structures, problems and solutions. The main reason is due to the fact the feasibility of an algorithm is usually measured in terms of the computation time as a function of the length of the input to the algorithm. Note that since the tally representation of a number is of exponential length in comparison to the binary representation, it follows that a function which is polynomial time computable in the tally representation of the natural numbers is not necessarily polynomial time computable in the binary representation of the natural numbers. Indeed, we can only conclude that such a function is exponential time computable in the binary representation. Thus it is essential that a definite representation be given for a feasible structure. A related reason is that two feasible sets need not be feasibly isomorphic. In particular, $Tal(\omega)$ and $Bin(\omega)$ are not p-time isomorphic. Thus we may have a p-time structure, say a graph, with universe $Tal(\omega)$, which is not isomorphic to a p-time structure with universe $Bin(\omega)$.

For a string $\sigma = (\sigma(0), \sigma(1), \dots, \sigma(n-1))$, $|\sigma|$ denotes the length n of σ . The empty string has length 0 and will be denoted by $\emptyset$. A constant string σ of length n consisting entirely of k 's will be denoted by k^n . For $m < |\sigma|$, $\sigma[m$ is the string $(\sigma(0), \dots, \sigma(m-1))$. We say σ is an *initial segment* of τ (written $\sigma \prec \tau$) if $\sigma = \tau[m$ for some m . The *concatenation* $\sigma \frown \tau$ (or sometimes just $\sigma\tau$) is defined by:

$$\sigma \frown \tau = (\sigma(0), \sigma(1), \dots, \sigma(m-1), \tau(0), \tau(1), \dots, \tau(n-1)),$$

where $|\sigma| = m$ and $|\tau| = n$. We write $\sigma \frown a$ for $\sigma \frown (a)$ and $a \frown \sigma$ for $(a) \frown \sigma$. For any $x \in \Sigma^\omega$ and any finite n , the *initial segment* $x[n$ of x is:

$$(x(0), \dots, x(n-1)).$$

We write $\sigma \prec x$ if $\sigma \prec x[n$ for some n . For any $\sigma \in \Sigma^n$ and any $x \in \Sigma^\omega$, we have $\sigma \frown x = (\sigma(0), \dots, \sigma(n-1), x(0), x(1), \dots)$. Given strings σ and τ of length n , we let

$$\sigma \otimes \tau = (\sigma(0), \tau(0), \dots, \sigma(n-1), \tau(n-1));$$

if $|\sigma| = n+1$ and $|\tau| = n$, then $\sigma \otimes \tau = (\sigma[n \otimes \tau) \frown \sigma(n)$. Given two elements x, y of Σ^ω , $x \otimes y = z$ where $z(2m) = x(m)$ and $z(2m+1) = y(m)$.

Strings may be coded by natural numbers in the usual fashion. Let $[x, y]$ denote the standard pairing function $\frac{1}{2}(x^2 + 2xy + y^2 + 3x + y)$, and in general

$$[x_0, \dots, x_n] = [[x_0, \dots, x_{n-1}], x_n].$$

Then a string σ of length n may be coded by

$$\langle \sigma \rangle = [n, [\sigma(0), \sigma(1), \dots, \sigma(n-1)]]$$

and also $\langle \emptyset \rangle = 1$.

Our basic computation model is the standard multitape Turing machine of Hopcroft and Ullman [68]. Note that there are different heads on each tape and that the heads are allowed to move independently. This implies that a string σ can be copied in linear time.

Let $t(n)$ be a function on natural numbers such that $t(n) \geq n$. A (deterministic) Turing machine M is said to be *$t(n)$ -time bounded* if each computation of M on inputs of size n where $n \geq 2$ is completed in at most $t(n)$

steps. If t can be chosen to be a polynomial then M is said to be *polynomial-time bounded*. A function $f(x)$ on strings is said to be in $\text{DTIME}(t)$ if there is a $t(n)$ -time bounded Turing machine M which computes $f(x)$. A set of strings or a relation on strings is in $\text{DTIME}(t)$ if its characteristic function is in $\text{DTIME}(t)$. We let

$$\begin{aligned} P &= \bigcup_i \{\text{DTIME}(n^i) : i \geq 0\}, \\ \text{DEXT} &= \bigcup_{c \geq 0} \{\text{DTIME}(2^{c \cdot n})\}, \\ \text{DEXPTIME} &= \bigcup_{c \geq 0} \{\text{DTIME}(2^{n^c})\}, \text{ and} \\ \text{DDOUBEXT} &= \bigcup_{c \geq 0} \{\text{DTIME}(2^{2^{c \cdot n}})\}. \end{aligned}$$

We say that a function $f(x)$ is *polynomial time* if $f(x) \in P$, is *exponential time* if $f(x) \in \text{DEXT}$, and is *double exponential time* if $f(x) \in \text{DDOUBEXT}$.

A function f is said to be *non-deterministic polynomial time* (NP) if there is a finite alphabet Σ , a polynomial p , a p -time relation R and a p -time function g such that, for any σ and τ ,

$$f(\sigma) = \tau \iff (\exists \rho \in \Sigma^{p(\sigma)}) [R(\rho, \sigma) \ \& \ g(\rho, \sigma) = \tau].$$

A *tree* T over Σ^* is a set of finite strings from Σ^* which contains the empty string $\emptyset$ and which is closed under initial segments. We say that $\tau \in T$ is an *immediate successor* of a string $\sigma \in T$ if $\tau = \sigma \frown a$ for some $a \in \Sigma$. Since our alphabet will always be countable and effective, we may assume that $T \subseteq \omega^{<\omega}$. Such a tree is said to be ω -*branching* since each node has potentially a countably infinite number of immediate successors. If each node of T has finitely many immediate successors, then T is said to be *finitely branching*. A recursive tree T is said to be *highly recursive* if there is a partial recursive function f such that, for any $\sigma \in T$, σ has at most $f(\sigma)$ immediate successors in T . For any tree T , an *infinite path* through T is a sequence $(x(0), x(1), \dots)$ such that $x[n] \in T$ for all n . Let $[T]$ be the set of infinite paths through T .

A subset X of ω^ω is a Π_1^0 *class* if $X = [T]$ for some recursive tree T . If the tree T is finitely branching, we will say that $P = [T]$ is a *bounded Π_1^0 class*, and if T is highly recursive, then $X = [T]$ is said to be a *recursively bounded*

(r.b.) Π_1^0 class. In Section eleven, we will define the notions of p-time and highly p-time trees and of p-time presented and p-time bounded Π_1^0 classes.

The terminology “ Π_1^0 ” indicates that a Π_1^0 class may be represented in arithmetic by a formula having one universal number quantifier. This alternate and perhaps more natural definition of a Π_1^0 class is discussed in Section 2.1.

Note that a Π_1^0 class $P \subseteq \omega^\omega$ is always a closed set and that, just as for the real line, P is compact if and only if P is closed and bounded. Thus a bounded Π_1^0 class is compact. Hence a recursively bounded Π_1^0 class may be thought of as being recursively compact.

Given two trees S and T contained in $\omega^{<\omega}$, we let

$$S \otimes T = \{\sigma \otimes \tau : \sigma \in S \text{ \& } \tau \in T \text{ \& } |\tau| \leq |\sigma| \leq |\tau| + 1\}.$$

For two Π_1^0 classes $P = [S]$ and $Q = [T]$, define the *amalgamation* of P and Q , $P \otimes Q$, by $P \otimes Q = \{x \otimes y : x \in P \text{ \& } y \in Q\}$. Then it is clear that $P \otimes Q = [S \otimes T]$. More generally, define the infinite amalgamation $\otimes_i S_i$ to be those strings σ such that for each i , $(\sigma([i, 0]), \sigma([i, 1]), \dots, \sigma([i, j])) \in S_i$, where j is the maximum such that $[i, j] < |\sigma|$. Then $[\otimes_i S_i]$ is isomorphic to the direct product $\prod_i [S_i]$.

We also wish to consider the following notion of disjoint union. Given two trees S and T contained in $\omega^{<\omega}$,

$$S \oplus T = \{\emptyset\} \cup \{0^\frown \sigma : \sigma \in S\} \cup \{1^\frown \tau : \tau \in T\}.$$

For two Π_1^0 classes $P = [S]$, $Q = [T]$, $P \oplus Q = \{0^\frown x : x \in P\} \cup \{1^\frown y : y \in Q\}$. It is easy to see that $[S \oplus T] = [S] \oplus [T]$. Clearly $S \oplus T$ is bounded if and only if both S and T are bounded, and similarly for the other notions of boundedness. More generally, the infinite disjoint union $\oplus_i T_i$ of trees is defined to be $\{\emptyset\} \cup \{(i) * \sigma : \sigma \in T_i\}$ and the infinite disjoint union $\oplus_i Q_i$ of classes is defined to be $\{(i) * y : y \in Q_i\}$.

A node σ of a tree T is said to be *extendible* if there is some $x \in [T]$ such that $\sigma \prec x$. The set of extendible nodes is denoted by $Ext(T)$ and a node $\sigma \in T$ is said to be a *dead end* if $\sigma \notin Ext(T)$.

It is important that any Π_1^0 class can be represented as the set $[T]$ of infinite paths through a primitive recursive (in fact, polynomial-time) tree. Thus the Π_1^0 classes can be effectively enumerated. This result will be proved in Section 2.2. The representation of a tree required to define the notion of a feasible tree will be given in Section 11.

We will be primarily interested in the two spaces $\{0,1\}^\omega$ (the Cantor space) and $\omega^\omega = \omega^*$ (the Baire space). The topology on ω^ω is determined by a basis of intervals $I(\sigma) = \{x : \sigma \prec x\}$. Notice that each interval is also a closed set and is therefore said to be *clopen* and that the clopen subsets of the Cantor space are just the finite unions of intervals.

The Cantor-Bendixson derivative $D(P)$ of a compact subset P of ω^ω is the set of nonisolated points of P . Thus a point $x \in P$ is not in $D(P)$ if and only if there is some open set U containing x which contains no other point of P . Equivalently, $x \notin D(P)$ if and only if there is some clopen set U such that $U \cap P = \{x\}$. Another useful observation is that, for any compact set P , $D(P)$ is empty if and only if P is finite.

The iterated Cantor-Bendixson derivative $D^\alpha(P)$ of a closed set P is defined for all ordinals α by the following transfinite induction.

$$D^0(P) = P;$$

$$D^{\alpha+1}(P) = D(D^\alpha(P)) \text{ for any } \alpha;$$

$$D^\lambda(P) = \bigcap_{\alpha < \lambda} D^\alpha(P) \text{ for any limit ordinal } \lambda.$$

The *Cantor-Bendixson rank* of a countable closed subset P of $\{0,1\}^\omega$ is the least ordinal α such that $D^{\alpha+1}(P) = \emptyset$. The (effective) Cantor-Bendixson rank of a point $x \in \{0,1\}^\omega$ is the least ordinal α such that, for some Π_1^0 class P , $D^\alpha(P) = \{x\}$.

We refer the reader to Rogers [131], Hinman [66], Soare [143], or Odifreddi [121] for the basic definitions of recursion theory. In particular we let φ_i be the partial recursive functional computed by the i -th Turing machine M_i . Given a string $\sigma \in \{0,1\}^*$, we write $\varphi_{i,s}(\sigma) \downarrow$ if M_i gives an output in s or fewer steps when started on input string σ . Thus the function $\varphi_{i,s}$ is uniformly polynomial time. We write $\varphi_e(\sigma) \downarrow$ if $(\exists s)(\varphi_{e,s}(\sigma) \downarrow)$ and $\varphi_e(\sigma) \uparrow$ if not $\varphi_e(\sigma) \downarrow$. Given two sets A and B , we write $A \leq_T B$ if A is Turing reducible to B and we write $A \equiv_T B$ if both $A \leq_T B$ and $B \leq_T A$. We let $\mathbf{0}$ denote the degree of the recursive sets and we let $\mathbf{0}'$ denote the degree of the jump of the recursive sets.

Let Γ be some complexity class of sets (and functions), such as partial recursive, primitive recursive, exponential time, polynomial time (or p-time). We say that a set or function is Γ -computable if it is in Γ .

We shall consider structures over an effective language:

$$\mathcal{L} = \langle \{R_i^{m(i)}\}_{i \in S}, \{f_i^{n(i)}\}_{i \in T}, \{c_i\}_{i \in U} \rangle,$$

where S , T and U are initial segments of ω , for all $i \in U$, c_i is a constant symbol and there are partial recursive functions s and t such that, for all $i \in S$, R_i is an $s(i)$ -ary relation symbol and, for all $i \in T$, f_i is a $t(i)$ -ary function symbol.

The language also includes variables and both existential and universal quantifiers using these variables. The set of terms of $\mathcal{L}$ and the set $Sent(\mathcal{L})$ of sentences of $\mathcal{L}$ are defined as usual by recursion. A propositional language is given by a set of 0-ary relation symbols, or propositional variables. The reader is referred to Shoenfield [138] for details.

A *model* or *structure*,

$$\mathcal{A} = (A, \{R_i^{\mathcal{A}}\}_{i \in S}, \{f_i^{\mathcal{A}}\}_{i \in T}, \{c_i^{\mathcal{A}}\}_{i \in U}),$$

for the language $\mathcal{L}$ is given by a set A together with interpretations of the relation, function and constant symbols.

Definition 1.1 A structure (where the universe A of $\mathcal{A}$ is a subset of Σ^*) is a Γ -*structure* if

- (i) A is a Γ -computable subset of Σ^*
- (ii) for each $i \in S$, $R_i^{\mathcal{A}}$ is a Γ -computable relation on $A^{m(i)}$.
- (iii) for each $j \in T$, $f_j^{\mathcal{A}}$ is a Γ -computable function from $A^{n(j)}$ into A .
- (iv) If $S = \omega$, then there is a Γ -computable relation R such that, for all $i \in S$ and all $(x_0, \dots, x_{m(i)})$,

$$R_i^{\mathcal{A}}(x_0, \dots, x_{m(i)}) \iff R(i, \langle x_0, \dots, x_{m(i)} \rangle).$$

- (v) If $T = \omega$, then there is a Γ -computable function f such that, for all $j \in T$ and all $(x_0, \dots, x_{n(j)})$,

$$f_j^{\mathcal{A}}(x_0, \dots, x_{n(j)}) = f(j, \langle x_0, \dots, x_{n(j)} \rangle).$$

For any complexity class Γ , we say that two structures $\mathcal{A}$ and $\mathcal{B}$ are Γ -isomorphic if there is an isomorphism f from $\mathcal{A}$ onto $\mathcal{B}$ and Γ -computable functions F and G such that $f = F \upharpoonright A$ (the restriction of F to A) and $f^{-1} = G \upharpoonright B$.

2 Π_1^0 classes and their members

In this section we will present results on Π_1^0 classes and their members which can be applied to the mathematical problems which will consider in the later sections. For the most part, we will have parallel results for arbitrary Π_1^0 classes, for bounded Π_1^0 classes, for recursively bounded Π_1^0 classes, and for Π_1^0 classes $P = [T]$ where T is a recursive tree with no dead ends. We also generalize some of these results to Π_n^0 classes by means of reducibility.

The section will be divided into several sub-sections. In Section 2.1, we discuss the alternate definition of a Π_1^0 class in terms of the arithmetic hierarchy and give a few fundamental lemmas such as Kleene's Normal Form Theorem for recursive functionals.

Section 2.2 contains the standard basis and anti-basis results on members of Π_1^0 classes. For example, we show that every r.b. Π_1^0 class contains an element of r.e. degree, while there exists an r.b. Π_1^0 class with no recursive member.

Section 2.3 will cover countable Π_1^0 classes, the idea of Cantor-Bendixson derivative and rank. For example, we show that any hyperarithmetic real is Turing equivalent to a member of a countable Π_1^0 class.

Section 2.4 will deal with the recent notions of *thin* and *minimal* Π_1^0 classes. Thus we construct a countable Π_1^0 class P such that for any Π_1^0 subclass Q of P ,

- (i) Q is either finite or is cofinite in P , and
- (ii) Q is the intersection of P with a clopen set.

In Section 2.5 we consider reducibility of Π_1^0 classes and of Π_n^0 classes. For example, we prove the Jockusch-Lewis-Remmel Theorem that any strong Π_2^0 class may be put in a one-to-one degree-preserving correspondence with a bounded Π_1^0 class.

The connection between reverse mathematics and the Π_1^0 class of separating sets for r.e. sets A, B will be considered in Section 2.6. The key result here is that if every class of r.e. separating sets can be represented by a problem, then the existence of a solution to this problem is equivalent to Weak König's Lemma in the subsystem RCA_0 of Peano Arithmetic.

We will consider in Section 2.7 an enumeration P_e of the Π_1^0 classes and various index sets associated with this enumeration, such as the set $\{e : P_e \text{ is nonempty}\}$, which is shown to be a complete Σ_1^1 set.

2.1 Π_1^0 classes and the Arithmetic Hierarchy

A recursive functional φ takes as inputs both numbers $a \in \omega$ and functions $x : \omega \rightarrow \omega$. The function inputs are treated as “oracles” to be called on when needed. Thus a particular computation

$$\varphi(a_1, \dots, a_n, x_1, \dots, x_m)$$

only uses a finite amount of information $x_i \upharpoonright c$ about each function x_i . This situation is covered by Kleene’s Normal Form Theorem (see [121], p. 180).

Lemma 2.1 *For each m, n , there exist a primitive recursive function U and a primitive recursive relation $T_{m,n}$ of numerical variables such that*

- (a) $\varphi_e(x_1, \dots, x_m, a_1, \dots, a_n) \downarrow$
 $\iff (\exists c) T_{m,n}(e, a_1, \dots, a_n, x_1 \upharpoonright c, \dots, x_m \upharpoonright c, c).$
- (b) $\varphi_e(x_1, \dots, x_m, a_1, \dots, a_n) \simeq U(c)$ if
 $T_{m,n}(e, a_1, \dots, a_n, x_1 \upharpoonright c, \dots, x_m \upharpoonright c, c).$

We have taken as our basic definition that a Π_1^0 class is the set of infinite paths through a recursive tree. Now the alternate definition from the viewpoint of the arithmetic hierarchy says that $P \subseteq \omega^\omega$ is a Π_1^0 class if there is a recursive relation R such that $x \in P \iff (\forall n) R(n, x)$.

Our next lemma demonstrates the equivalence of the quantifier definition of Π_1^0 classes with the tree definition as well as with a slightly more general notion as the set of infinite paths through a Π_1^0 tree. We give the relativized version.

Lemma 2.2 *For any class $P \subseteq \omega^\omega$ and any $z \in \omega^\omega$, the following are equivalent:*

- (a) $P = [T]$, for some tree $T \subseteq \omega^{<\omega}$ recursive in z .
- (b) $P = [T]$, for some tree T primitive recursive in z .
- (c) $P = \{x : (\forall n) R(n, x)\}$, for some relation R recursive in z .
- (d) $P = [T]$, for some tree $T \subseteq \omega^{<\omega}$ which is Π_1^0 in z .

Proof. For simplicity of presentation we omit the parameter z .

[(a) $\rightarrow$ (b)]: Suppose that $P = [T]$, where T is a recursive tree and let φ_e be a total $\{0, 1\}$ -valued recursive function such that $\sigma \in T$ if and only if $\varphi_e(\sigma) = 1$. Define the primitive recursive tree S by

$$\tau \in S \iff (\forall n < |\tau|) \neg \varphi_{e, |\tau|}(\tau \upharpoonright n) = 0.$$

Clearly $T \subseteq S$, so that $[T] \subseteq [S]$. Suppose now that $x \notin [T]$. Then for some n , $x \upharpoonright n \notin T$. Thus we have some m such that $\varphi_{e, m}(x \upharpoonright n) = 0$. Then for any $k > \max\{m, n\}$, we clearly have $x \upharpoonright k \notin S$. It follows that $x \notin [S]$.

[(b) $\rightarrow$ (c)]: Suppose that $P = [T]$ where T is a primitive recursive tree. Define the relation R by

$$R(n, x) \iff x \upharpoonright n \in T.$$

Then we have

$$x \in [T] \iff (\forall n) x \upharpoonright n \in T \iff (\forall n) R(n, x).$$

[(c) $\rightarrow$ (d)]: Suppose that $x \in P \iff (\forall n) R(n, x)$ where R is a recursive relation, that is, there is a recursive functional $\varphi = \varphi_e$ such that $R(n, x) \iff \varphi(n, x) = 1$ and $\neg R(n, x) \iff \varphi(n, x) = 0$. Define the tree T by

$$\sigma \in T \iff (\forall k, c < |\sigma|) [T_{1,1}(e, n, \sigma \upharpoonright c, c) \rightarrow U(c) = 1].$$

It is clear that $P = [T]$.

[(d) $\rightarrow$ (a)]: Suppose that the tree T is a Π_1^0 subset of $\omega^{<\omega}$ so that there is a recursive relation R such that

$$\sigma \in T \iff (\forall n) R(n, \sigma).$$

Define the recursive tree $S \supset T$ by

$$\sigma \in S \iff (\forall m, n \leq |\sigma|) R(m, \sigma \upharpoonright n).$$

It is easily verified that $[T] = [S]$. □

We will also consider the families of Π_n^0 classes. Recall that a subset A of ω is said to be Π_n^0 if there exists a recursive relation R and an alternating sequence of number quantifiers beginning with $(\forall a_1)$ such that

$$a \in A \iff (\forall a_1)(\exists a_2) \cdots (Q_n a_n) R(a, a_1, \dots, a_n).$$

Similarly, a subclass P of ω^ω is said to be a Π_n^0 class if there exists a recursive relation R and an alternating sequence of number quantifiers beginning with $(\forall a_1)$ such that

$$P(x) \iff (\forall a_1)(\exists a_2) \cdots (Q_n a_n) R(x, a_1, \dots, a_n).$$

As usual, a Σ_n^0 class is simply the complement of a Π_n^0 class and a Δ_n^0 class is both Π_n^0 and Σ_n^0 . It is well known that a class is Δ_{n+1}^0 if and only if it is recursive in the Σ_n^0 -complete set $\mathbf{0}^{(n)}$, the n -th jump of the empty set.

In particular, P is said to be a *strong* Π_n^0 class if there is a Π_n^0 tree T such that $P = [T]$. It follows from the usual contraction of quantifiers that a strong Π_n^0 class is in fact a Π_n^0 class. For strong Π_n^0 classes there is one addition to Lemma 2.2.

Lemma 2.3 *For any class $P \subseteq \omega^\omega$ and any natural number n , the following are equivalent:*

- (a) P is a strong Π_{n+1}^0 class;
- (b) $P = [T]$, for some tree $T \subseteq \omega^{<\omega}$ recursive in $\mathbf{0}^{(n)}$;
- (c) $P = [T]$, for some tree $T \subseteq \omega^{<\omega}$ which is Σ_n^0 .

Proof. The equivalence of (a) and (b) follows from Lemma 2.2 with $z = \mathbf{0}^{(n)}$. Clearly (c) implies (a). It remains to be shown that (a) implies (c). Let T be a Π_{n+1}^0 tree such that $P = [T]$. Then there is a Σ_n^0 relation R such that

$$\sigma \in T \iff (\forall i) R(i, \sigma),$$

so that

$$x \in P \iff (\forall m)(\forall i) R(i, x \upharpoonright m).$$

Now define the Σ_n^0 tree S by

$$\sigma \in S \iff (\forall m \leq |\sigma|)(\forall i \leq |\sigma|) R(i, \sigma \upharpoonright m).$$

Since T is a tree, it is clear that for $\sigma \in T$ and $m \leq |\sigma|$, we have $\sigma \upharpoonright m \in T$, so that $\sigma \in S$. Thus $T \subseteq S$, so that $[T] \subseteq [S]$.

Next suppose that $x \in [S]$ and let $\sigma = x \upharpoonright m$ for some m . For any i , let $k = \max\{m, i\}$. Then $x \upharpoonright k \in S$ which implies that $R(i, \sigma)$. Thus $x \upharpoonright m \in T$ for each m and hence $x \in [T]$. Thus $[S] \subseteq [T]$. $\square$

For more details on the arithmetic hierarchy, see Chapter 4 of Soare [143].

Recall that a set A enumerated in increasing order as $a_0, a_1, \dots$ is said to be *hyperimmune* if for any recursive function f , there is an n such that $a_n > f(n)$ and A is said to be *retraceable* if there is a partial recursive function φ such that $\varphi(a_{n+1}) = a_n$ for all n . It is a theorem of Dekker and Myhill [33] that any retraceable, non-recursive Π_1^0 set is hyperimmune.

Here are three natural examples of Π_1^0 classes associated with sets of natural numbers. For any function $f : \omega \rightarrow \omega$, the set

$$K(f) = \{x : (\forall n)(x(n) \leq f(n))\}$$

defines a compact subset of ω^ω . For any infinite set A , the *principal function* p_A enumerates the elements of A in increasing order. If A is an infinite Π_1^0 set, then $K(p_A)$ is a Π_1^0 class since $x \in K(p_A)$ if and only if, for each n , there exist no more than n members of A which are $\leq x(n)$.

For any Π_1^0 set C , $\mathcal{P}(C) = \{x : (\forall n)(x(n) = 1 \rightarrow n \in C)\}$ is a Π_1^0 class. If C is finite, then $\mathcal{P}(C)$ is also finite. However if C is infinite, then $\mathcal{P}(C)$ is uncountable and perfect. For an infinite set $C = \{c_0 < c_1 < \dots\}$, consider also the family $I(C)$ of *initial subsets* of C where the initial subsets are those subsets F for which $c_{n+1} \in F$ implies $c_n \in F$ for all n . It is shown in [17] that $I(C)$ is a Π_1^0 class if and only if the set C is *co-r.e.* and is retraceable. It is clear that $I(C)$ is always countable and contains at most one non-recursive member (C).

2.2 Members of Π_1^0 classes

For a given domain X , such as $X = \omega$ or $X = \omega^\omega$, we say that a class $\Gamma \subseteq X$ of is a *basis* for a family Θ of classes if every nonempty class from Θ has a member from Γ . For example, the class of recursive functions is a basis for the family of open subclasses of ω^ω . This is an example of a “basis theorem”. An antibasis theorem has the opposite form, that is, that there exists some class in Θ which has no member from Γ . For example, the class of recursive functions is not a basis for the family of closed subclasses of ω^ω since every singleton is a closed class.

In this section, we consider several basis and antibasis theorems.

The classical result here is König’s Infinity Lemma (König [90]) which states that any infinite, finitely branching tree has an infinite branch. Recall that any tree T has a subtree $Ext(T)$ of infinitely extendible nodes. The most

fundamental basis theorem is the following result of Kleene. The effective version of König's Lemma depends upon the complexity of the tree $\text{Ext}(T)$.

Lemma 2.4 *For any tree T such that the Π_1^0 class $P = [T]$ is nonempty, P contains a member which is recursive in $\text{Ext}(T)$.*

Proof. The infinite path x through T can be defined recursively by letting $x(0)$ be the least n such that $(n) \in \text{Ext}(T)$ and, for each $k \geq 0$, letting $x(k+1)$ be the least n such that $(x(0), \dots, x(k), n) \in \text{Ext}(T)$. $\square$

Lemma 2.5 *For any recursive tree $T \subseteq \omega^{<\omega}$:*

- (a) $\text{Ext}(T)$ is a Σ_1^1 set;
- (b) if T is finitely branching, then $\text{Ext}(T)$ is a Π_2^0 set;
- (c) if T is highly recursive, then $\text{Ext}(T)$ is a Π_1^0 set.

Proof.

(a) In general,

$$\sigma \in \text{Ext}(T) \iff (\exists x)(\forall n > |\sigma|)[x \upharpoonright n \in T \ \& \ \sigma \prec x \upharpoonright n].$$

(b) If T is finitely branching, then König's Lemma implies that

$$\sigma \in \text{Ext}(T) \iff (\forall n > |\sigma|)(\exists \tau)[|\tau| = n \ \& \ \sigma \prec \tau \ \& \ \tau \in T].$$

(c) Finally, suppose that T is highly recursive and let f be a recursive function such that $\sigma(i) < f(i)$ for all $\sigma \in T$ and all $i < |\sigma|$. Then the quantifier " $\exists \tau$ " in (b) is bounded, since $\tau(i) < f(i)$ for all $i < n$. $\square$

Combining Lemmas 2.4 and 2.5, we get the following.

Theorem 2.6 *For any nonempty Π_1^0 class $P \subseteq \omega^\omega$,*

- (a) P has a member recursive in some Σ_1^1 set;
- (b) if P is bounded, then P has a member recursive in $\mathbf{0}''$;
- (c) if P is recursively bounded, then P has a member recursive in $\mathbf{0}'$;
- (d) if $P = [T]$, where T has no dead ends, then P has a recursive member.

Part (a) of this theorem is the Kleene basis theorem [84] and part (c) is the Kreisel basis theorem [91].

Jockusch and Soare [75] obtained several important refinements of Theorem 2.6(c), some of which we collect together in the following theorem. They also showed in [74] that any r.b. Π_1^0 class contains a hyperimmune member a such that $\mathbf{a}'' = \mathbf{0}''$. Note also that Shoenfield [137] showed first that any nonempty r.b. Π_1^0 class P contains a member a which has degree $\mathbf{a} <_T \mathbf{0}'$.

Theorem 2.7 *Let P be a nonempty bounded Π_1^0 class. Then,*

- (a) *P contains a member of Σ_2^0 degree and, if P is r.b., then P contains a member of r.e. degree.*
- (b) *P contains a member a such that $a' \leq_T \mathbf{0}''$ and, if P is r.b., then P contains a member a such that $\mathbf{a}' = \mathbf{0}'$.*
- (c) *P contains members a and b such that any function recursive in both a and b is recursive in $\mathbf{0}'$ and, if P is r.b., then P contains members a and b such that any function recursive in both a and b is recursive.*

Proof. It will be shown in Theorem 2.23 below that if P is bounded, then it is bounded by some function f recursive in $\mathbf{0}'$. We omit the proof of (c) here.

(a) It will be shown in Section 9 below that the minimal element x of a bounded Π_1^0 class P has Σ_2^0 degree and that if P is r.b., then x has r.e. degree.

(b) We give the proof for the r.b. case indicated in the exercise from Soare [143, p. 109]. We may assume from Theorem 2.21 below that P is in fact a binary class. Recall that $x' = \{e : \varphi_e^x(e) \downarrow\}$. Let $P = [T]$, where T is primitive recursive, let $T_0 = T$ and define, recursively in $\mathbf{0}'$, a sequence $T_\pi(e)$ of primitive recursive subtrees of T as follows. For each e , define the primitive recursive tree

$$U_e = \{\sigma \in \{0, 1\}^{<\omega} : \varphi_{e, |\sigma|}^\sigma(e) \uparrow\}.$$

Let $T_{\pi(e+1)} = T_{\pi(e)}$ if $T_{\pi(e)} \cap U_e$ is finite and let $T_{\pi(e+1)} = T_{\pi(e)} \cap U_e$ otherwise. Then $[T_\pi(e)]$ is nonempty for each e and $[T_{\pi(e+1)}] \subseteq [T_{\pi(e)}]$ so that $\bigcap_e [T_{\pi(e)}] \neq \emptyset$ by compactness. Now let $x \in \bigcap_e [T_{\pi(e)}]$. Then $x \in P$ and we claim that $x' = \mathbf{0}'$.

We observe that the function π is recursive in $\mathbf{0}'$, since $T_e \cap U_e$ is finite if and only if it has no members of length n for some n , that is, if and only if

$$(\exists n)(\forall \sigma \in \{0, 1\}^n)[\varphi_{\pi(e)}(\sigma) = 1 \rightarrow \varphi_{e, |\sigma|}^\sigma(e) \downarrow].$$

Observe that for each e , the tree $T_{\pi(e+1)}$ has the property that either $\varphi_e^u(e) \downarrow$ for all $u \in [T_{e+1}]$ or $\varphi_e^u(e) \uparrow$ for all $u \in T_{e+1}$. Thus $e \in x'$ if and only if $T_e \cap U_e$ is finite, which can be checked recursively in $\mathbf{0}'$, as seen above. It follows that x' is recursive in $\mathbf{0}'$, so that $\mathbf{x}' = \mathbf{0}'$.

If the tree $P = [T]$ is bounded by a function recursive in $\mathbf{0}'$, then we may assume from Theorem 2.24 that T is in fact a binary tree which is recursive in $\mathbf{0}'$. The argument given above relativizes to give x' recursive in $\mathbf{0}''$. $\square$

The second part of (b) is known as the Low Basis Theorem, since a is said to be *low* if $\mathbf{a}' = \mathbf{0}'$.

We observe that the family of Σ_1^1 classes has the same basis as the family of Π_1^0 classes. Clearly a basis for Σ_1^1 is always a basis for Π_1^0 . For the other direction, let S be a nonempty subset of ω^ω and suppose that $x \in S \iff (\exists y)P(x, y)$, where P is Π_1^0 . Then by Theorem 2.7, P has a member (x, y) recursive in some Σ_1^1 set and hence it follows that S has a member (x) which is likewise recursive in a Σ_1^1 set.

We now turn to some anti-basis results, showing that Theorem 2.7 is, in a reasonable sense, best possible. Part (a) is due to Kleene [86], part (b) is due to Jockusch-Lewis-Remmel [70], and part (c) is due to Kreisel [91].

Theorem 2.8

- (a) *There is a nonempty Π_1^0 class with no Δ_1^1 member.*
- (b) *There is a nonempty bounded Π_1^0 class with no element recursive in $\mathbf{0}'$.*
- (c) *There is a nonempty recursively bounded Π_1^0 class with no recursive member.*

Proof.

- (a) It follows from the theory of hyperarithmetic sets that

$$\{x : x \text{ is not } \Delta_1^1\}$$

is a Σ_1^1 class (see Hinman [66, p. 106]). The result is immediate.

(b) In Jockusch-Lewis-Remmel [70], it is shown that, for any strong Π_2^0 class Q , there is a bounded Π_1^0 class P and a one-to-one degree-preserving correspondence between P and Q . This is Theorem 2.24 (b) below. Now the relativized version of (c) gives us a strong Π_2^0 class Q with no member recursive in $\mathbf{0}'$ and the corresponding bounded Π_1^0 class P likewise has no member recursive in $\mathbf{0}'$.

(c) The proof is a simple diagonal argument. As usual, let $\varphi_{e,s}$ be the computation of φ_e after s steps. Define the recursive tree $T \subseteq \{0, 1\}^{<\omega}$ by

$$\sigma = (x(0), x(1), \dots, x(n-1)) \in T$$

if and only if, for each $e < n$, it is not the case that $\varphi_{e,n}(e) = x(e)$. Now $P = [T]$ is nonempty, since $x \in P$, where

$$x(e) = \begin{cases} 1 - \varphi_e(e), & \text{if } \varphi_e(e) \downarrow, \\ 0, & \text{if } \varphi_e(e) \uparrow. \end{cases}$$

For any $x \in P$ and any e such that φ_e is total, it is clear that $x \neq \varphi_e$, since if $\varphi_{e,s}(e) \downarrow$, then $x[s \in T$ implies that $x(e) \neq \varphi_e(e)$. $\square$

We next consider Π_1^0 classes with no recursive members. Such classes were called *special* by Jockusch and Soare in [75, 74]. It was shown in [75] that for any special (possibly unbounded) Π_1^0 class P , there is a non-zero r.e. degree $\mathbf{a}$ such that P has no members of degree $\leq \mathbf{a}$. Thus if P has members of every r.e. degree, then P must have a recursive member.

Jockusch and Soare give several constructions of special r.b. Π_1^0 classes in [75, 74]. We summarize these in the following. Part (e) is due to Kucera [93].

Theorem 2.9

- (a) *There is an r.b. Π_1^0 class P such that for any degree $\mathbf{a}$ of a member of P and any r.e. degree $\mathbf{b} \geq_T \mathbf{a}$, $\mathbf{b} = \mathbf{0}'$.*
- (b) *For any r.e. degree $\mathbf{c}$, there is an r.b. Π_1^0 class P such that the r.e. degrees of members of P are exactly those $\geq_T \mathbf{c}$.*
- (c) *For any degree $\mathbf{a}$, there is a nonempty r.b. Π_1^0 class P with no recursive members and with no member of degree $\mathbf{a}$.*

- (d) *There is a nonempty r.b. Π_1^0 class P such that any two members of P are Turing incomparable.*
- (e) *There is a nonempty r.b. Π_1^0 class P such that, for any degree $\mathbf{a} \leq_T \mathbf{0}'$ of a member of P , there is a non-recursive r.e. degree $\mathbf{b}$ with $\mathbf{b} \leq_T \mathbf{a}$.*

Recently, Groszek and Slaman [62] constructed an r.b. perfect Π_1^0 class P such that every member of P either has r.e. degree or has minimal degree.

It follows from Theorem 2.12 below that any special r.b. Π_1^0 class must be perfect. This was improved in [75] by Jockusch and Soare to the following.

Theorem 2.10 *For any r.b. Π_1^0 class P with no recursive members and any countable set $\{\mathbf{a}_i : i < \omega\}$ of nonrecursive degrees, P has a continuum of pairwise Turing incomparable members x such that the degree of x is Turing incomparable with each $\mathbf{a}_i$. Moreover, P has members a, b such that $\mathbf{a} \wedge \mathbf{b} = \mathbf{0}$.*

Jockusch and Soare also constructed in [75] an r.b. Π_1^0 class P (which happens to be the class of subsets of a Π_1^0 set) such that

$$\mathcal{D}(P) = \{x : (\exists y \in P) (x \equiv_T y)\}$$

has Lebesgue measure 1. On the other hand, they showed that even for an arbitrary (possibly unbounded) Π_1^0 class P , the set $\mathcal{D}(P)$ is meager.

By the results of Jockusch-Lewis-Rommel in Theorem 2.24, the results of Theorem 2.9 may be relativized with a $\mathbf{0}'$ oracle to obtain the following.

Theorem 2.11

- (a) *There is a strong Π_2^0 class P of sets such that for any degree $\mathbf{a}$ of a member of P and any Σ_2^0 degree $\mathbf{b} \geq_T \mathbf{a}$, $\mathbf{b} =_T \mathbf{0}''$.*
- (b) *For any Σ_2^0 degree $\mathbf{c}$, there is a strong Π_2^0 class P of sets such that the Σ_2^0 degrees of members of P are exactly those $\geq_T \mathbf{c}$.*
- (c) *For any degree $\mathbf{a} \geq_T \mathbf{0}'$, there is a strong Π_2^0 class P of sets with no members recursive in $\mathbf{0}'$ and with no member of degree $\mathbf{a}$.*
- (d) *There is a nonempty strong Π_2^0 class P of sets such that for any two degree $\mathbf{a}, \mathbf{b}$ of members of P , $\mathbf{a} \not\leq_T \mathbf{b} \vee \mathbf{0}'$.*
- (e) *There is a nonempty strong Π_2^0 class of sets P such that, for any degree $\mathbf{a} \leq \mathbf{0}''$ of a member of P , there is a Σ_2^0 degree $\mathbf{b}$ with $\mathbf{0}' <_T \mathbf{b} \leq_T \mathbf{a}$.*

2.3 Countable Π_1^0 classes

We begin with the basis theorem of Kreisel [92].

Theorem 2.12 *Let P be a Π_1^0 class.*

- (a) *Any isolated member of P is hyperarithmetical; if P is finite, then every member of P is hyperarithmetical.*
- (b) *Suppose that P is bounded. Then any isolated member of P is recursive in $\mathbf{0}'$. If P is finite, then every member of P is recursive in $\mathbf{0}'$.*
- (c) *Suppose P is recursively bounded. Then any isolated member of P is recursive. If P is finite, then every member of P is recursive.*

Proof. Let x be an isolated element of $P = [T]$. Then we can find an n such that $P \cap I(x \upharpoonright n) = \{x\}$. Now define the recursive subtree S of T to be $\{\sigma : x \upharpoonright n \text{ is compatible with } \sigma\}$. Then x is the only element of $[S]$. Thus without loss of generality we may assume that x is the unique element of $P = [T]$. Then $\text{Ext}(T) = \{x \upharpoonright n : n < \omega\}$ so that x is clearly recursive in $\text{Ext}(T)$. We use the fact that $\text{Ext}(T)$ contains exactly one sequence $(x \upharpoonright n)$ of length n , for each n . Thus, for each n , every other sequence of length n is a dead end. We now consider the three cases above.

- (a) For an (unbounded) tree T , $\text{Ext}(T)$ is Σ_1^1 by Lemma 2.5 and we have

$$\sigma \in \text{Ext}(T) \iff (\forall \tau \in \omega^{|\sigma|}) (\tau \neq \sigma \rightarrow \tau \notin \text{Ext}(T)),$$

so that $\text{Ext}(T)$ is also Π_1^1 . It follows that $\text{Ext}(T)$ and hence x are hyperarithmetical.

(b) For a finitely branching tree T , $\text{Ext}(T)$ is Π_2^0 . Now it follows from König's Lemma that for each n , there is some $k \geq n$ such that every sequence in T of length k is an extension of $x \upharpoonright n$. Thus we have

$$\sigma \in \text{Ext}(T) \iff (\exists k \geq |\sigma|) (\forall \tau \in \omega^k) (\tau \in T \rightarrow \sigma \prec \tau),$$

so that $\text{Ext}(T)$ is also Σ_2^0 . It follows that $\text{Ext}(T)$ and hence x are recursive in $\mathbf{0}'$.

(c) For a highly recursive tree T with recursive bounding function f , $\text{Ext}(T)$ is Π_1^0 and we have $\sigma \in \text{Ext}(T)$ if and only if

$$(\forall \tau \in \{0, 1, \dots, \max(f(0), \dots, f(|\sigma|))\}^{|\sigma|}) (\tau \neq \sigma \rightarrow \tau \notin \text{Ext}(T)),$$

so that $\text{Ext}(T)$ is also Σ_1^0 . It follows that $\text{Ext}(T)$ and hence x are recursive.

Suppose now that P is finite. The conclusion in each case follows from the fact that every member of P will be isolated. $\square$

Theorem 2.13 *Any countable Π_1^0 class has an isolated point.*

Proof. We prove the contrapositive. Suppose that P has no isolated points. Then $D(P) = P$, that is, P is dense in itself. Now P is also closed and therefore perfect. But it is a classical result (see Moschovakis [114, p. 16]) that any perfect set must contain a copy of the Cantor space $\{0, 1\}^\omega$ and is therefore uncountable. $\square$

Corollary 2.14 *Let P be a countable Π_1^0 class.*

- (a) *P has a hyperarithmetical member.*
- (b) *Suppose that P is bounded. Then P has a member recursive in $\mathbf{0}'$.*
- (c) *Suppose P is recursively bounded. Then P has a recursive member.*

The more general study of countable Π_1^0 classes is accomplished through a generalization of the notion of an isolated point. The classical Cantor-Bendixson theorem states that any closed subset P of the Cantor set $\{0, 1\}^\omega$ is the union of a perfect set $K(P)$, called the *perfect kernel* of P , with a countable set S . Define the *Cantor-Bendixson rank* $rk_P(x)$ in P of an element x in P to be the least ordinal α such that $x \in D^\alpha(P) \setminus D^{\alpha+1}(P)$. The Cantor-Bendixson rank $\alpha(P)$ of a closed set is the least ordinal α such that $D^\alpha(P) = D^{\alpha+1}(P)$ and the (perfect) kernel $K(P) = D^{\alpha(P)}(P)$. $K(P)$ is perfect since $D(K(P)) = K(P)$. Finally, set $S = P \setminus K(P)$.

We see that $\alpha(P)$ and S are countable as follows. Recall that any infinite compact subset Q of $\{0, 1\}^\omega$ has a limit point x such that every neighborhood of x contains infinitely many elements of Q . Moreover if Q is uncountable, then it has a *strong limit point* such that every neighborhood of x contains uncountably many elements of Q . It follows that $D^\alpha \setminus D^{\alpha+1}$ is countable for every α so that S is uncountable if and only if $\alpha(P)$ is uncountable. Now suppose that $\alpha(P)$ were uncountable. Then $P \setminus D^{\omega_1}(P)$ would be uncountable and therefore have a strong limit point x . It follows that, for each countable α , every neighborhood of x would contain an element of D^α ,

which implies that $x \in D^{\alpha+1}(P)$. Thus $x \in D^{\omega_1}(P)$, which contradicts the choice of x and shows that $\alpha(P)$ is countable. It follows as above that $K(P)$ consists of exactly the strong limit points of P .

For a Π_1^0 class P , we can say more. We first show that if $x \in P \setminus K(P)$, then we can find a countable Π_1^0 class $Q \subseteq P$ such that $x \in Q$. To define Q , just take an interval I such that x is isolated in $I \cap D^\alpha(P)$ and let $Q = I \cap P$. Then it is easy to see that $D^\alpha(Q) = \{x\}$, so that $K(Q) = \emptyset$ and Q must be countable.

Kreisel [92] used the Boundedness Principle of Spector to show that for a Π_1^0 class P , $\alpha(P) \leq \omega_1^{\text{CK}}$, where ω_1^{CK} is the least non-recursive ordinal (named after Church and Kleene), and that $K(P)$ is a Σ_1^1 set. He also showed that every element of a countable recursively bounded Π_1^0 class is hyperarithmetical and that the degrees of members of such classes are cofinal in the hyperarithmetical degrees. These results were obtained through an analysis of a Π_1^0 class by means of the Cantor-Bendixson derivative and were refined by Cenzer, Clute, Smith, Soare and Wainer in [16, 28]. Recall that the *Cantor Bendixson rank* $rk(x)$ is $\min\{rk_P(x) : P \text{ is a } \Pi_1^0 \text{ class}\}$. The following result is from [16].

Theorem 2.15 *Let $x \in \{0, 1\}^\omega$, let T be a highly recursive tree, $P = [T]$ an r.b. Π_1^0 class, n a natural number and λ a recursive limit ordinal. Then,*

- (a) *If $rk_P(x) = n$, then x is recursive in $T^{(2n)}$; furthermore, if T has no dead ends, then for $n > 0$, x is recursive in $T^{(2n-1)}$.*
- (b) *If $rk_P(x) = \lambda + n$, then x is recursive in $T^{(\lambda+2n)}$.*
- (c) *$rk_P(x)$ is a recursive ordinal for any $x \in P$.*

Proof. Recall the Π_1^0 definition of $\text{Ext}(T)$, the set of nodes of T which have an infinite extension in $[T]$. The proof is based on the fact that $D(P) = [d(T)]$, where the tree $d(T)$ is recursive in $T^{(2)}$ and is defined by

$$s \in d(T) \iff (\exists t)[s \prec t \text{ \& } t \cap 0 \in \text{Ext}(T) \text{ \& } t \cap 1 \in \text{Ext}(T)].$$

Now we can iterate the operator d to inductively define a monotone decreasing sequence of trees T_α for every recursive ordinal α and a tree $T^* = \cap_\alpha T_\alpha$ such that $D^\alpha(P) = [T_\alpha]$ and $K(P) = [T^*]$. The Boundedness Principle implies that this sequence cannot continue past the recursive ordinals. (See

Hinman [66] or Cenzer and Mauldin [20] for a detailed discussion of inductive definability and the Boundedness Principle.) Now it is easy to see from the definition of the operator d that the tree T_α is recursive in $T^{(2\alpha)}$. If T has no dead ends, then $T = \text{Ext}(T)$ so that $d(T)$ is recursive in T' . For details, see (Hinman [66, p. 148]). $\square$

It is immediate from Theorem 2.15 that every element of a countable recursively bounded Π_1^0 class is hyperarithmetic. The problem of determining which hyperarithmetic sets can have rank α in a countable Π_1^0 class was studied in [16, 28, 17]. We summarize the results below in Theorems 2.16 and 2.17.

Recall the class $I(C)$ of initial subsets of a retraceable Π_1^0 set C . It is easy to see that the Cantor-Bendixson derivative $D(I(C)) = \{C\}$, so that C has rank one in $I(C)$. It follows that any retraceable Π_1^0 set has effective Cantor-Bendixson rank one.

Theorem 2.16

- (a) For any r.e. set A , there is a retraceable Π_1^0 set $B \equiv_T A$ with rank 1.
- (b) For any Δ_2^0 set A , there is a hyperimmune set $B \equiv_T A$ with $\text{rk}(B) = 1$.
- (c) For any finite n and any set A such that $\mathbf{0}^{(2n-1)} \leq_T A \leq_T \mathbf{0}^{(2n)}$ or $\mathbf{0}^{(2n-2)} \leq_T A \leq_T \mathbf{0}^{(2n-1)}$, there is a set $B \equiv_T A$ with $\text{rk}(B) = n$.
- (d) For any finite n , any limit ordinal λ and any set A such that A is r.e. in $\mathbf{0}^{(\lambda+2n+1)}$ and $\mathbf{0}^{(\lambda+2n+1)} \leq_T A$, there is a set $B \equiv_T A$ such that $\text{rk}(B) = \lambda + n + 1$.

Proof.

(a) This is Theorem 2.8 of [28]. We will sketch the proof here because it is a nice illustration of the idea of rank. First observe that any r.e. set A is Turing equivalent to a retraceable Π_1^0 set B , obtained from A as follows. Let A be recursively enumerated without repetition as $a_0, a_1, \dots$. Then we define a sequence $b_0, b_1, \dots$ by induction as follows. First let $b_0 = m$ where a_m is the least element of A . Then let b_n be the least $b > b_{n-1}$ (for $n > 0$) such that for all $a \leq n$, $a \in A \iff (\exists i < b)(a = a_i)$. It is easy to see that B is retraceable, Π_1^0 and Turing equivalent to A . Then B has rank one in the Π_1^0 class $I(B)$ as observed above.

(b) This is Corollary 2.2 of [28, p. 981] and the proof is a modification of the proof of (a).

(c) This is Corollary 3.2 of [16, p. 163].

(d) This is Corollary 2.7 of [16, p. 160]. □

Cenzer and Remmel [26] recently improved part (d) by replacing the condition “ A is r.e. in $0^{\lambda+2n+1}$ ” to “ $A \leq_T 0^{\lambda+2n+1}$ ”. The existence of sets which do not have rank, or which have high rank and low degree, was explored by Cenzer and Smith in [28]. An important fact here is that two sets may have the same Turing degree but different rank. We summarize the results of [28] in the following theorem together with the result of Jockusch and Shore [72] which constructs a totally unranked degree.

Theorem 2.17

- (a) *For any recursive ordinal α , $0^{(\alpha)}$ is not ranked.*
- (b) *For each recursive ordinal α , there is a Δ_2^0 set with rank α .*
- (c) *For any nonrecursive r.e. degree $\mathbf{a}$, there is an r.e. set B of degree $\mathbf{a}$ which is not ranked.*
- (d) *For any degree $\mathbf{a} \geq_T \mathbf{0}'$, there is a set B of degree $\mathbf{a}$ which is not ranked.*
- (e) *There is a Σ_2^0 degree $\mathbf{a}$ such that no set of degree $\mathbf{a}$ is ranked.*

Downey [37] improved (e) by finding a completely unranked hyperimmune-free degree and also found a degree $\mathbf{a} \leq_T \mathbf{0}''$ such that every degree $\leq \mathbf{a}$ contains a ranked set. Cholak and Downey [30] improved (b) by finding r.e. sets of arbitrary recursive rank.

2.4 Thin and minimal classes

Recent work on the lattice of Π_1^0 subclasses of $\{0, 1\}^\omega$ has involved the notions of *thin* and *minimal* Π_1^0 classes. An infinite Π_1^0 class P is said to be *thin* if, for every Π_1^0 subclass Q of P , there is a clopen set U such that $Q = U \cap P$. The first construction of such a class was due to Martin and Pour-El [109],

who constructed an axiomatizable, essentially undecidable theory T such that each extension of T was principal. This theory T can be viewed as a (perfect) thin Π_1^0 class. The notion of thinness was first made explicit by Downey [35]. Perfect thin classes were also constructed by Simpson and are related to superminimal profinite groups by the work of R. Smith [140]. An infinite Π_1^0 class C is said to be *minimal* if every Π_1^0 subclass Q of C is either finite or cofinite in C . Thus the notion of a minimal Π_1^0 class is the analog of the notion of a co-maximal Π_1^0 subset of ω . In particular, if C is a co-maximal set, then the class of subsets of C containing either one or no elements is an example of a minimal Π_1^0 class which is not thin. There are several other interesting connections between maximal r.e. sets and Π_1^0 classes in [17]. It is shown that a maximal r.e. set may not have rank one in a Π_1^0 class, but may have rank 2 and that there is a maximal r.e. set which is not a member of any thin Π_1^0 class. Downey and Yang [45] constructed a cohesive set of rank one.

There are several results in [17] relating the degree of a member x of a thin class to the Cantor-Bendixson rank of x . The most basic result is that a member x of a thin Π_1^0 class P is isolated in P if and only if x is recursive. To see this, note that if x is a recursive element of P , then $\{x\}$ is a Π_1^0 subclass of P and therefore $\{x\} = P \cap U$ for some clopen set U , which immediately implies that x is isolated in P . The connection between minimal and thin classes is that if P is thin and $D(P)$ is a singleton, then P is minimal, whereas if P is minimal and has a non-recursive member, then P is thin (and $D(P)$ is a singleton). Countable thin Π_1^0 classes are constructed in [17] of every recursive Cantor-Bendixson rank.

We state here the results which we will apply in later sections. First there are the results which apply to any thin Π_1^0 class.

The following Lemma is Theorem 2.13 of [17, p. 102]. The full hypothesis of this theorem was not stated in [17], so we take the opportunity here to correct that omission.

Lemma 2.18 *Let T be a recursive tree and P a Π_1^0 class such that $P = [T]$. Then for any set $A \in P$,*

- (a) *if $P \subseteq \mathcal{P}(A)$, then $A \leq_T \text{Ext}(T)$, and*
- (b) *if A is a Π_1^0 set and P is thin, then $A \leq_T \text{Ext}(T)$.*

Proof. The proof of (a) is given in [17].

Now for (b), note that $\mathcal{P}(A)$ is a Π_1^0 class, so that $Q = P \cap \mathcal{P}(A)$ is a Π_1^0 subclass of P and is nonempty since $A \in Q$. Since P is thin, we must have $Q = P \cap U$ for some clopen $U = I(\sigma_0) \cup \cdots \cup I(\sigma_k)$. Then set

$$T_Q = \{\sigma \in T : \sigma \text{ is compatible with } \sigma_i, \text{ for some } i \leq k\}.$$

It is clear that $Q = [T_Q]$ and that

$$\text{Ext}(T_Q) = \{\sigma \in \text{Ext}(T) : \sigma \text{ is compatible with } \sigma_i, \text{ for some } i \leq k\}.$$

Thus $\text{Ext}(T_Q)$ is recursive in $\text{Ext}(T)$. Now $Q \subseteq \mathcal{P}(A)$, so that by (a) we have $A \leq_T \text{Ext}(T_Q) \leq_T \text{Ext}(T)$. $\square$

Cholak, Coles and Downey [29] recently obtained a precise characterization of the degrees of $\text{Ext}(T)$ when $[T]$ is thin and perfect, namely the ANR degrees of Downey, Jockusch and Stob [41], extending the previous result of Downey [36] that all ANR degrees contain such classes.

The next result consists of Theorem 2.10 and Corollary 2.14 of [17].

Theorem 2.19 *Let T be a recursive tree such that $P = [T]$ is a thin Π_1^0 class and let $A \in P$. Then,*

- (a) $A' \leq_T A \oplus \mathbf{0}''$ (so that it is not possible that $A \geq_T \mathbf{0}''$).
- (b) If T has no dead ends, then $A' \leq_T A \oplus \mathbf{0}'$ (so that it is not possible that $A \geq_T \mathbf{0}'$).
- (c) If T has no dead ends and A is either r.e. or co-r.e., then A is recursive.

It follows from (b) and Theorem 2.16 (a) above that if A has rank one in a thin Π_1^0 class $P = [T]$, where T has no dead ends, then A has *low* degree $\mathbf{a}$, that is, $\mathbf{a}' = \mathbf{0}'$.

In contrast to Theorem 2.19 (a), it is shown in Theorem 2.18 of [17] that there is a minimal thin Π_1^0 class P and a set A such that $D(P) = \{A\}$ and $A \oplus \mathbf{0}' \equiv_T \mathbf{0}''$.

The next result consists of Theorems 2.2, 2.9 and 2.16 of [17].

Theorem 2.20

- (a) *For any recursive ordinal α , there is a thin Π_1^0 class P_α of Cantor-Bendixson rank α such that P_α is the set of paths through a tree with no dead ends.*
- (b) *There is an r.e. A set of degree $\mathbf{0}'$ and a minimal, thin Π_1^0 class P with $D(P) = \{A\}$.*
- (c) *Between any two distinct r.e. degrees, there is a degree $\mathbf{a}$, a set A of degree $\mathbf{a}$ and a minimal, thin Π_1^0 class P with $D(P) = \{A\}$.*

On the other hand, there is an r.e. degree $\mathbf{a}$ and a minimal degree $\mathbf{b}$ such that no set of degree $\mathbf{a}$ or $\mathbf{b}$ belongs to any thin Π_1^0 class. Thus we have degrees of which some of the members belong to thin Π_1^0 classes and degrees of which none of the members belong to thin Π_1^0 classes. Finally, there is a degree $\mathbf{a} \leq \mathbf{0}''$ such that every set of degree $\leq \mathbf{a}$ belongs to a thin Π_1^0 class.

2.5 Reducibility of classes

In this section we study various notions of reducibility between bounded and unbounded Π_1^0 classes, between strong Π_2^0 classes and Π_1^0 classes, and between Π_n^0 classes and Π_1^0 classes.

The connection between unbounded Π_1^0 classes and r.b. Π_1^0 classes was studied by Jockusch and Soare in [74] and by Cenzer and Smith in [28]. The connection between Π_2^0 classes and Π_1^0 classes was studied by Jockusch and McLaughlin [71] and by Jockusch, Lewis and Remmel in [70].

One basic goal is to reduce arbitrary Π_1^0 classes to classes of sets, that is, classes contained in $\{0, 1\}^\omega$. Some definitions are needed. There is a recursive map k from ω^ω to $\{0, 1\}^\omega$ such that $k(x)$ has the same Turing degree as x for all $x \in \omega^\omega$, defined by

$$k(x) = (0^{x(0)} \ 1 \ 0^{x(1)} \ 1 \ \dots).$$

Note that $\{k(x) : x \in P\}$ need not be a closed set, since P may not be compact.

For $t \in \omega^{<\omega}$ with $|t| = n$, let

$$k(t) = (0^{t(0)} \ 1 \ 0^{t(1)} \ 1 \ \dots \ 0^{t(n-1)} \ 1).$$

For a tree $T \subseteq \omega^{<\omega}$, the tree $k(T) \subseteq \{0, 1\}^\omega$ is defined by

$$k(T) = \{k(t)0^i : t \in T \text{ \& } i \in \omega\}.$$

Then for a Π_1^0 class $P = [T] \subseteq \omega^\omega$, define

$$k(P) = [k(T)] = \{k(x) : x \in [T]\} \cup \{k(t)0^\omega : t \in T\}.$$

This observation proves part (a) of the next theorem. Note that $k(P)$ always has recursive members. If P is recursively bounded, then this construction can be modified to obtain a homeomorphism between P and $k(P)$. This is part (b) below.

Let us say that a tree T *highly recursive in x* if T is recursive in x and there is a function f recursive in x such that $\sigma(n) \leq f(n)$ for all $\sigma \in T$. We will then say that the class $[T]$ is *bounded recursively in x* . In particular, we note that if T is a finite-branching Σ_1^0 tree, then T is highly recursive in $\mathbf{0}'$.

Theorem 2.21

- (a) *For any Π_1^0 class $P \subseteq \omega^\omega$, there is a Π_1^0 class $Q \subseteq \{0, 1\}^\omega$ and a one-to-one degree-preserving correspondence between the non-recursive members of P and the non-recursive members of Q .*
- (b) *Any r.b. Π_1^0 class is recursively homeomorphic to a Π_1^0 class of sets.*
- (c) *Any Π_2^0 class P which is highly recursive in $\mathbf{0}'$ is recursively homeomorphic to a strong Π_2^0 class of sets.*

Proof.

(b) Let $T \subseteq \omega^{<\omega}$ be a recursive tree, let $P = [T]$, and let f be a recursive function such that for all $\tau \in T$ and all $n < |\tau|$, $\tau(n) < f(n)$. Now define the tree $S \subseteq \{0, 1\}^{<\omega}$ to be:

$$S = \{k(\tau) \frown 0^i : \tau \in T \text{ \& } i < f(|\tau|)\}.$$

It is then clear that k is a recursive homeomorphism from P onto $[S]$.

(c) This is just a relativization to $\mathbf{0}'$ of the proof of (b). □

Jockusch and Soare showed in Theorem 1 of [74] that an arbitrary Π_1^0 class P with no recursive members can be represented by an r.b. Π_1^0 class Q in the sense that the degrees of members of P are a subset of the degrees of members of Q . We give this result together with a relativized version.

Theorem 2.22

- (a) For any Π_1^0 class $P \subseteq \omega^\omega$ with no recursive members, there is a Π_1^0 class R of sets with no recursive members such that $\mathcal{D}(P) \subseteq \mathcal{D}(R)$.
- (b) For any Π_1^0 class $P \subseteq \omega^\omega$ with no members recursive in $\mathbf{0}'$, there is a strong Π_2^0 class R of sets with no members recursive in $\mathbf{0}'$ such that $\mathcal{D}(P) \subseteq \mathcal{D}(R)$.

Proof.

(a) Let Q be a Π_1^0 class of sets with no recursive member. Let $P = [S]$, let $Q = [T]$ and assume without loss of generality that $S \subseteq (\omega \setminus \{0, 1\})^{<\omega}$.

It clearly suffices, by Theorem 2.21 (b) to obtain a class R which is recursively bounded and otherwise meets the requirements of the conclusion. Define the tree U to be the set of strings

$$\mu = \sigma_1 * \tau_1 * \sigma_2 * \tau_2 * \cdots * \sigma_n * \tau_n$$

such that $\sigma_i \neq \emptyset$ for $i > 1$, $\tau_j \neq \emptyset$ for $j < n$,

$$s(\mu) = \sigma_1 * \cdots * \sigma_n \in S,$$

$\tau_j \in T$ for all j , and $\mu(k) \leq k + 1$ for all k .

We claim that $R = [U]$ satisfies the requirements of the theorem.

The tree U is finite-branching by the restriction that $\mu(k) \leq k + 1$. Thus R is a recursively bounded Π_1^0 class.

Now for any $x \in P$ we can define $z \in R$ with the same degree as x as follows. First of all, define a recursive sequence $\emptyset = t_0, t_1, \dots$ such that t_j is the lexicographically least string in T of length j . Now given x , let

$$z = t_{i_0} * (x(0)) * t_{i_1} * (x(1)) \cdots,$$

where for each n , i_n is the least such that

$$x(n) \leq |t_{i_0} * (x(0)) * \cdots * t_{i_{n-1}} * x(n-1) * t_{i_n}| + 1.$$

Then z is recursive in x by the definition, and x is recursive in z , since it is the subsequence of z consisting of the entries $z(n) > 1$.

It remains to be shown that R has no recursive members. Let $z \in R$. There are two cases.

CASE 1: Suppose that $z(i) > 1$ for infinitely many i and let $i_0, i_1, \dots$ enumerate $\{i : z(i) > 1\}$. Define x by $x(n) = z(i_n)$. It is clear that x is recursive in z and that $x \in P$. Hence x is not recursive and therefore z is not recursive.

CASE 2: Suppose that $z(i) > 1$ for only finitely many i and let m be the largest such that $z(m) > 1$. Define y by $y(n) = z(m+n)$. It is clear that y is recursive in z and that $y \in Q$. Hence y is not recursive and therefore z is not recursive.

(b) The proof is just a modification of the proof of (a). Let $Q = [T]$ in this case be a strong Π_2^0 class of sets with no member recursive in $\mathbf{0}'$ (by Theorem 2.8). Then we define a tree U recursive in $\mathbf{0}'$ with $\mu(k) \leq k+1$ for all $\mu \in U$, so that $R = [U]$ is an r.b. strong Π_2^0 class with the desired properties and apply Theorem 2.21 (c). $\square$

In particular we see that any member of a countable Π_1^0 class $P \subseteq \omega^\omega$ is Turing equivalent to a member of a countable Π_1^0 class $Q \subseteq \{0,1\}^\omega$. The Π_1^0 singletons in ω^ω are of particular interest. It is well-known that any hyperarithmetic set is Turing reducible to a Π_1^0 singleton (see Hinman [66, p. 153]). It was shown by Clote [31] that $0^{(\alpha)}$ is Turing equivalent to a Π_1^0 singleton for any recursive ordinal α . Clote defines a version of rank for Π_1^0 singletons x called the *height* of x , which is the supremum of the heights of the dead ends in the tree T such that $\{x\} = [T]$. Cenzer-Smith [28] used the correspondence of Theorem 2.21 to establish a relationship between the height of a Π_1^0 singleton x in ω^ω and the Cantor-Bendixson rank of the image $\varphi(x)$. This is used to show for example that any Δ_2^0 function x is Turing equivalent to a Π_1^0 singleton y with height ω [28, p. 986].

These results on Π_1^0 singletons demonstrate the distinction between bounded and general Π_1^0 classes. This is because if $\{x\}$ is a bounded Π_1^0 set, then x must be recursive in $\mathbf{0}'$ by Corollary 2.14. Thus, for example, if x has degree $\mathbf{0}''$ and $P = \{x\}$ is a Π_1^0 class, then there can be no bounded Π_1^0 class Q with a one-to-one degree-preserving correspondence between P and Q . This shows that parts (a) and (c) of Theorem 2.21 and Theorem 2.22 (a) are best possible in that the restriction to “non-recursive members” cannot be removed from part (a) of Theorem 2.21 and in that the inclusion $\mathcal{D}(P) \subseteq \mathcal{D}(Q)$ of Theorem 2.22 cannot be improved to equality.

We next consider some further questions about bounded Π_1^0 classes. Recall the definition above that a Π_1^0 class P is bounded if $P = [T]$ for some

recursive, finitely branching tree T . There are two possible alternative definitions. One would be that there exists a recursive tree T and a (bounding) function f such that $P = [T]$ and such that $\sigma(n) \leq f(n)$ for all $\sigma \in T$. A second alternative would be that there exists a recursive tree T and a function f such that $P = [T]$ and $x(n) \leq f(n)$ for all $x \in P$. Certainly the first alternative implies the second. We show below that this second alternative is equivalent to the notion of compactness. However, this alternative is not equivalent to our definition since by the remarks above we may have $P = \{x\}$ for x of arbitrarily high hyperarithmetic degree, whereas a bounded Π_1^0 singleton must be recursive in $\mathbf{0}'$ by Theorem 2.12 (b).

Theorem 2.23

- (a) A Π_1^0 class P is bounded if and only if there exists a recursive tree T and a function f such that $\sigma(n) \leq f(n)$ for all $\sigma \in T$. Furthermore, f may be taken to be recursive in $\mathbf{0}'$.
- (b) A Π_1^0 class P is compact if and only if there exists a function f such that $x(n) \leq f(n)$ for all $x \in P$.
- (c) For any recursive tree T with no dead ends, $P = [T]$ is compact if and only if P is a bounded Π_1^0 class.

Proof.

(a) If $P = \emptyset$, then this is obvious. Thus we let P be a nonempty Π_1^0 class and let T be a recursive tree such that $P = [T]$. Suppose first that T is finitely branching. Then we may define the bounding function f by letting $f(n)$ be the maximum of $\{\sigma(n) : \sigma \in T\}$. It is clear that f is recursive in $\mathbf{0}'$. Conversely, suppose that $P = [T]$ and that f is any bounding function such that $\sigma(n) \leq f(n)$ for all $\sigma \in T$. It is immediate that T must be finitely branching.

(b) Suppose first that P is compact. For each n , $P \subset \bigcup_i \{x : x(n) = i\}$, and it follows from compactness that there exists some i_n such that $P \subseteq \{x : x(n) \leq i_n\}$. Then the function $f(n) = i_n$ satisfies the condition above. Suppose next that there exists such a bounding function f . Then $P \subseteq \prod_{n \in \omega} \{0, 1, \dots, f(n)\}$ and is therefore compact since it is a closed subset of $\prod_{n \in \omega} \{0, 1, \dots, f(n)\}$ which is compact by the classical Tychonoff Theorem, being a product of discrete, finite compact spaces.

(c) It follows from (b) that any bounded Π_1^0 class is compact. Now let T be a recursive tree with no dead ends, and suppose that $P = [T]$ is compact, that is, there exists a bounding function f such that $x(n) \leq f(n)$. Then it is clear that f is also a bounding function for T , so that P is bounded by (a). $\square$

We next present the Jockusch-Lewis-Remmel Theorem from [70]. Recall that P is a strong Π_2^0 class if there exists a tree recursive in $\mathbf{0}'$ such that $P = [T]$. Let us say that P is a bounded strong Π_2^0 class if there is a tree T which is recursive in $\mathbf{0}'$ and a bounding function f also recursive in $\mathbf{0}'$ such that $P = [T]$ and such that $\sigma(n) \leq f(n)$ for all $\sigma \in T$. Thus in particular any strong Π_2^0 class of sets is a bounded Π_2^0 class.

Theorem 2.24

- (a) *For any bounded Π_1^0 class Q , there exists a bounded strong Π_2^0 class of sets P and an effective one-to-one degree-preserving correspondence between P and Q .*
- (b) *For any bounded strong Π_2^0 class P , there is a bounded Π_1^0 class Q and an effective one-to-one degree-preserving correspondence between P and Q .*

Proof.

(a) Let $P = [T]$, where T is a finitely branching, recursive tree. It follows from Theorem 2.23(a) that T has a bounding function f which is recursive in $\mathbf{0}'$. Now define the tree S , recursive in $\mathbf{0}'$, as in part (b) of Theorem 2.22, to obtain a strong Π_2^0 class $[S]$ of sets such that the function $k(x)$ is an effective one-to-one degree-preserving map from P onto $[S]$.

(b) Let $P = [S]$, where S is highly recursive in $\mathbf{0}'$. Then as in the proof of Theorem 2.21(c), we may assume that T is a binary tree. It now follows from Lemma 2.3 that T may be assumed to be a Σ_1^0 tree. Thus there is a recursive relation R such that

$$x \in P \iff (\forall m)(\exists n) R(n, x \upharpoonright m).$$

Now we may define Q by

$$z = x \otimes y \in Q \iff (\forall m)[R(y(m), x \upharpoonright m) \ \& \ \forall i < y(m) (\neg R(i, x \upharpoonright m))].$$

Then for each $x \otimes y \in Q$, we have $x \in P$ and for each $x \in P$, there is a unique y such that $x \otimes y \in Q$ and that y is defined so that $y(m)$ is the least n such that $R(n, x[m])$. Thus y is recursive in x and therefore $x \otimes y$ has the same degree as x . $\square$

We note that it is a corollary to the proof above that any bounded Π_1^0 class is a strong bounded Π_2^0 class.

Combining Theorems 2.22 (b) and 2.24 (b), we obtain the following.

Corollary 2.25 *For any Π_1^0 class P with no member recursive in $\mathbf{0}'$, there is a bounded Π_1^0 R class with no members recursive in $\mathbf{0}'$ such that $\mathcal{D}(P) \subseteq \mathcal{D}(R)$.*

The same technique can be extended to arbitrary Π_n^0 classes in the following result essentially due to Jockusch and McLaughlin [71], Theorem 3.1 and Remark 3.4. This will allow us to extend some of the results about Π_1^0 classes to Π_2^0 classes.

Theorem 2.26 *For any natural number n and any Π_n^0 class P , there is a Π_1^0 class Q and an effective one-to-one degree-preserving correspondence between the members of P and the members of Q .*

Proof. It clearly suffices to prove that for any n and any Π_{n+2}^0 class P , there is a Π_{n+1}^0 class Q and an effective one-to-one degree-preserving correspondence between the members of P and the members of Q .

Suppose that we have a Π_n^0 relation R such that

$$x \in P \iff (\forall m)(\exists n) R(m, n, x).$$

Now let

$$x \otimes y \in Q \iff (\forall m) [R(m, y(m), x) \ \& \ (\forall i < y(m)) (\neg R(m, i, x))].$$

Then Q is a Π_{n+1}^0 class and the result follows as in the proof of Theorem 2.24. $\square$

We observe that the effective degree-preserving correspondences in Theorems 2.24 and 2.26 are in fact recursive homeomorphisms. By this we mean that a homeomorphism H from P onto Q is recursive if there exist partial recursive functionals F and G such that the $H = F \upharpoonright P$ and $H^{-1} = G \upharpoonright Q$.

It is not hard to see that it is not the case that every strong Π_2^0 class can be put in a one-to-one degree-preserving correspondence with an r.b. Π_1^0 class. For example, if A is recursive in $0'$ but not recursive, then $\{A\}$ is a strong Π_2^0 class. If $\{A\}$ were in one-to-one correspondence with an r.b. Π_1^0 class $\{B\}$, then B would have to be recursive by Theorem 2.12 so that the correspondence could not be degree-preserving.

Thus if we are to have a reduction of an arbitrary strong Π_2^0 class P to a Π_1^0 class Q , we must either drop the requirement that Q is recursively bounded, as was done in Theorem 2.24, weaken the requirement of degree-preserving, or weaken the requirement of a one-to-one correspondence and make φ an embedding of P into Q . Two results are given below. Consider again the problem where A is recursive in $0'$ but non-recursive and φ is an embedding of $\{A\}$ into an r.b. Π_1^0 class Q . Then $\varphi(A)$ may not both have the same degree as A and the same rank (0) in Q that A has in $\{A\}$.

The following theorem from Cenzer-Smith [28, p. 986] shows that we can preserve degree if we do not preserve rank. It also shows that the rank of a set in a strong Π_2^0 class is at most one lower than the rank of A is a Π_1^0 class.

Theorem 2.27 *For any strong Π_2^0 class $P \subset \{0,1\}^\omega$, there is a Π_1^0 class $Q \subseteq \{0,1\}^\omega$ and a recursive isomorphism φ from P into Q such that, for all A in P , $rk_Q(\varphi(A)) \leq 1 + rk_P(A)$.*

We can preserve rank if we do not preserve degree. The following result is an immediate corollary of Theorem 3.1 from Cenzer-et-al [16, p. 161].

Theorem 2.28 *For any strong Π_3^0 class $P \subset \{0,1\}^\omega$, there is a Π_1^0 class $Q \subseteq \{0,1\}^\omega$ and a homeomorphism φ from P onto $D(Q)$ such that, for all $A \in P$, $x \leq_T \varphi(x) \leq x \oplus 0'$.*

2.6 Separating Sets

An important example of Π_1^0 classes in recursion theory is the class of separating sets of a pair of disjoint r.e. sets. If A and B are infinite disjoint r.e. sets, then C is a *separating set* for A and B if $A \subseteq C$ and $B \cap C = \emptyset$. We let $S(A, B)$ denote the class of separating sets for A and B . In particular, $S(A, \emptyset)$ is the class of supersets of A and $S(\emptyset, B)$ is the class $\mathcal{P}(\{0,1\}^\omega \setminus B)$ of subsets of the Π_1^0 set $\{0,1\}^\omega \setminus B$. Also $S(A, B) = S(A, \emptyset) \cap S(\emptyset, B)$. A and B are said to be *recursively inseparable* if there is no recursive separating

set C for A and B . This concept was introduced by Kleene in [85], where recursively inseparable r.e. sets were constructed. Shoenfield showed in [136] that every non-recursive r.e. degree contains a pair of recursively inseparable sets. Shoenfield observed in [137] that $S(A, B)$ is an r.b. Π_1^0 class. Observe that $S(A, B)$ is finite if and only if $A \cup B$ is cofinite, in which case A and B are both recursive and every separating set is also recursive. Otherwise, $S(A, B)$ is a perfect set and thus has the cardinality of the continuum. In either case, both the r.e. set A and the co-r.e. set $\omega \setminus B$ are of course separating sets for A and B . The results above yield the following additional conclusions.

Theorem 2.29 *For any pair A, B of recursively inseparable disjoint r.e. sets:*

- (a) *There are separating sets C and D for A and B such that any function recursive in both C and D is recursive.*
- (b) *There is a non-zero r.e. degree $\mathbf{a}$ such that no separating set C for A and B is recursive in $\mathbf{a}$.*
- (c) *For any countable set $\{\mathbf{a}_i : i < \omega\}$ of nonrecursive degrees, there are continuum pairwise Turing incomparable, separating sets C for A and B which are incomparable with each $\mathbf{a}_i$.*

Theorem 2.30 *For any two r.e. sets A and B , $S(A, B)$ is not thin.*

Proof. If $A \cup B$ is cofinite, then $S(A, B)$ is finite and therefore not thin. Otherwise, define the Π_1^0 class Q by

$$C \in Q \iff C \in S(A, B) \ \& \ (\forall i, j) [(i \in C \wedge j \in C) \rightarrow (i \in A \vee j \in A)].$$

That is, Q contains the set A along with those separating sets of the form $A \cup \{i\}$. Now Q is a Π_1^0 subclass of $S(A, B)$, so that if P were thin, then we would have a clopen set $U = I(\sigma_0) \cup \dots \cup I(\sigma_k)$ such that $Q = S(A, B) \cap U$. Since $A \in Q$, it follows that $A \in U$, so that there is some finite n such that for any C , if

$$C \cap \{0, 1, \dots, n\} = A \cap \{0, 1, \dots, n\}$$

and $C \in S(A, B)$, then $C \in Q$. Since $A \cup B$ is not cofinite, we can choose two elements $b, c > n$ with $b, c \notin A \cup B$. It would follow that $A \cup \{b, c\} \in Q$, contradicting the definition of Q . $\square$

As indicated above, the class of separating sets of A and B can never be a countable set and therefore cannot represent an arbitrary Π_1^0 class. However, parts (c) and (d) of Theorem 2.9 above can be done with classes of separating sets. These results are from [75, p. 48].

Theorem 2.31

- (a) *For any degree $\mathbf{a}$, there exist recursively inseparable r.e. sets A and B which have no separating set of degree $\geq_T \mathbf{a}$.*
- (b) *There exist recursively inseparable r.e. sets A and B such that any two separating sets C, D for A and B are Turing incomparable unless the symmetric difference of C, D is finite.*

The concept of an *array non-recursive* (a.n.r.) r.e. set, introduced by Downey, Jockusch and Stob [41] is important in the study of separating sets. It is shown in [41] that an r.e. set C is a.n.r. if and only if there exist r.e. sets A and B , each recursive in A , such that no set of degree $\mathbf{0}'$ separates A and B .

The problem of finding a separating set for a pair of disjoint r.e. sets has been studied in connection with reverse mathematics. Reverse mathematics is the program of Harvey Friedman, S. Simpson and others to answer the question: *What set existence theorems are needed to prove ordinary theorems of mathematics?* König's Lemma states that an infinite, finitely branching tree has an infinite path. Thus the existence of an infinite path in an arbitrary r.b. Π_1^0 class is equivalent to König's Lemma in a certain subsystem RCA_0 of arithmetic. In fact, it was shown by Simpson in Lemma 2.6 of [139] that the existence of a separating set for an arbitrary pair of disjoint r.e. sets also is equivalent to König's Lemma in RCA_0 .

2.7 Index Sets

A very important topic in recursion theory which is related to Π_1^0 classes is the notion of an *index set*. For an introduction to index sets for r.e. sets, see Soare [143]. A complete discussion of index sets for Π_1^0 classes is given by Cenzer and Remmel in [26]. Index sets for recursive trees were studied by Lempp in [95] in connection with the lattice of r.e. sets. Beigel and Gasarch [9, 10] and Gasarch and Martin [58] have recently studied index sets for various combinatorial problems, such as graph coloring, to measure the

complexity of these problem; they refer to certain index sets as “promise problems”. See the survey paper [57] by Gasarch for the application of index sets in many areas of combinatorics. Thurber [147] has studied index sets for Boolean algebras to strengthen earlier work of Feiner [49]. We will define and classify several index sets for Π_1^0 classes and show that they can play an important role in the application of Π_1^0 classes to problems in combinatorics and other areas of mathematics.

Recall that φ_a is the partial recursive function with index a and that W_a is the domain of φ_a (when φ_a is a function of one variable — otherwise $W_a = \emptyset$). A set $A \subset \omega$ is said to be an index set if for any a, b , $a \in A$ and $\varphi_a = \varphi_b$ imply that $b \in A$. Thus in particular, $\emptyset$ and ω are index sets. Rice’s Theorem states that these are the only two recursive index sets. For example, $K = \{a : a \in W_a\}$ is not an index set. However all of the following are index sets.

- (i) $K_1 = \{a : W_a \neq \emptyset\}$
- (ii) $Fin = \{a : W_a \text{ is finite}\}$
- (iii) $Inf = \{a : W_a \text{ is infinite}\}$
- (iv) $Cof = \{a : W_a \text{ is cofinite}\}$
- (v) $Coinf = \{a : W_a \text{ is coinfinite}\}$
- (vi) $Rec = \{a : W_a \text{ is a recursive set}\}$
- (vii) $Tot = \{a : \varphi_a \text{ is total}\}$
- (viii) $Ext = \{a : \varphi_a \text{ is extendible to a total recursive function}\}$
- (ix) $Comp = \{e : W_e \equiv K\}$
- (x) $U_1^1 = \{a : (\exists x)(\forall n)(x \upharpoonright n) \notin W_a\}$

We are particularly interested in the complexity of such index sets. Recall that a subset A of ω is said to be Σ_n^m -complete (respectively, Π_n^m -complete) if A is Σ_n^m (respectively, Π_n^m) and if any Σ_n^m (respectively, Π_n^m) set B is many-one reducible to A , where B is many-one reducible to A if there is a recursive function f such that, for any b , $b \in B$ if and only if $f(b) \in A$. More generally, the double completeness $(\Sigma_n^m, \Pi_n^m) \leq_m (B, C)$ for a disjoint pair of sets B and C means that, for some Σ_n^m complete set A , there is a recursive function f such that, for any a , $a \in A \iff f(a) \in B$ and $a \notin A \iff f(a) \in C$. If B is Σ_n^m , C is Π_n^m and $(\Sigma_n^m, \Pi_n^m) \leq_m (B, C)$, then we will say that the pair (B, C) is (Σ_n^m, Π_n^m) complete. For example, (Fin, Inf) is (Σ_2^0, Π_2^0) complete.

The index sets described above all turn out to be complete for some level of the arithmetical hierarchy. Here is a brief list of such complexity results, most taken from Soare [143] where the reader can find a further discussion of index sets. The last result can be found in Hinman [66, p. 84].

Theorem 2.32

- (i) K and K_1 are Σ_1^0 -complete sets.
- (ii) Fin is a Σ_2^0 -complete set.
- (iii) Inf and Tot are Π_2^0 -complete sets.
- (iv) Cof , Ext and Rec are Σ_3^0 -complete sets.
- (v) $Coinf$ is a Π_3^0 -complete set.
- (vi) $Comp$ is a Σ_4^0 -complete set.
- (vii) U_1^1 is a Σ_1^1 -complete set.

It follows from Lemma 2.2 that there are several possible approaches to enumerating the Π_1^0 classes. For the sake of simplicity, we use an indexing based on primitive recursive trees. Let $\pi_0, \pi_1, \dots$ be an effective enumeration of the primitive recursive functions from ω to $\{0, 1\}$ and let

$$U_e = \{\emptyset\} \cup \{\sigma : (\forall \tau \preceq \sigma) \pi_e(\langle \tau \rangle) = 1\}.$$

It is clear that each U_e is a primitive recursive tree. Observe also that if $\{\sigma : \pi(\sigma) = 1\}$ is a primitive recursive tree, then U_e will be that tree. Thus every primitive recursive tree occurs in our enumeration U_e . Then we let $P_e = [U_e]$ be the e -th Π_1^0 class. It follows from Lemma 2.2 that every Π_1^0 class occurs in the enumeration P_e .

There are a number of natural properties of Π_1^0 classes whose index sets we shall classify. We will consider whether a Π_1^0 class is nonempty and, if it is nonempty, whether it has a recursive element (or has an element of some given complexity). That is, we shall determine the complexity of the set of indices for nonempty Π_1^0 classes and the set of indices of Π_1^0 classes which have a recursive member. More generally, we consider index sets for various cardinality properties, e.g., whether a Π_1^0 class has exactly k elements, is finite or is countable. We also consider recursive cardinality properties, such

as having exactly k recursive members or having infinitely many recursive members. Other properties studied include measure-theoretic and topological properties, such as being meager or being co-meager, having measure $< r$ or $\geq r$ for a fixed real r , as well as set-theoretic properties, such as being a class of separating sets, being a thin class, or being a minimal class. Index sets related to higher order derivatives are also considered.

Once we have classified a variety of index sets for Π_1^0 classes, we can immediately transfer these results to index sets for various recursive version of mathematical problems, given that the set of solutions to such a recursive version of the mathematical problem is always a Π_1^0 class and that the set of solutions to the recursive version of the mathematical problem can strongly represent any arbitrary (bounded, r.b.) Π_1^0 class. A number of examples of such results can be found in [26]. For example, in Section 6 we will show that the index set of primitive recursive graphs having a 4-coloring is a Π_1^0 complete set, whereas the index set of primitive recursive graphs having a recursive 4-coloring is a Σ_3^0 complete set. This is a natural strengthening of the well-known result of Bean [7] that k -colorable recursive graphs need not have recursive k -colorings.

We will now give a selection of results from [26] with some indication of proofs. The first family of index sets is based on the notions of boundedness for Π_1^0 classes which are crucial in determining the complexity of the elements.

For any property $\mathcal{R}$ of trees, let $I_P(\mathcal{R})$ be the set of indices e such that U_e has the property $\mathcal{R}$. Thus for example, $I_P(\text{bounded}) = \{e : U_e \text{ is bounded}\}$.

Our first results will deal with index set for various notions of boundedness. Recall or a given function $g : \omega^{<\omega} \rightarrow \omega$, a tree $T \subseteq \omega^{<\omega}$ is said to be *g -bounded* if for every $\sigma \in \omega^{<\omega}$ and every $i \in \omega$, if $\sigma \frown i \in T$, then $i < g(\sigma)$. Thus, for example, if $g(\sigma) = 2$ for all σ , then a g -bounded tree is simply a binary tree. T is said to be *finitely branching* if T is g -bounded for some g , that is, if each node of T has finitely many immediate successors. Observe that this is equivalent to the existence of a bounding function h such that $\sigma(i) < h(i)$ for all $\sigma \in T$ and all $i < |\sigma|$. T is said to be *recursively bounded* (r.b.) if it is g -bounded for some recursive function g . As above, this is equivalent to the existence of a recursive bounding function h such that $\sigma(i) < h(i)$ for all $\sigma \in T$ and all $i < |\sigma|$. If T is recursive, then this is also equivalent to the existence of a partial recursive function f such that, for any $\sigma \in T$, σ has at most $f(\sigma)$ immediate successors in T . A recursive tree T is *highly recursive* if it is also recursively bounded.

In [26], Cenzer and Remmel introduced a new type of boundedness condition for trees. We will say that $T \subseteq \omega^{<\omega}$ is *almost bounded by* $g : \omega^{<\omega} \rightarrow \omega$ if there is some $k \in \omega$ such that for all σ with $|\sigma| > k$ and for all i , if $\sigma \smallfrown i \in T$, then $i < g(\sigma)$. T is said to be *almost bounded* (a.b.) if it is almost bounded by some g and *almost recursively bounded* (a.r.b.) if it is almost bounded by some recursive function g . Note that these notions are not equivalent to the existence of a (recursive) function h and a finite k such that $\sigma(i) < h(i)$ for all $\sigma \in T$ with $|\sigma| > k$ and all $i < |\sigma|$, as seen by the example of the tree $T = \{n^k : n, k \in \omega\}$, which is a.r.b. but has, for each k and n , strings $\sigma = n^{k+1} \in T$ with $\sigma(k) = n$.

Theorem 2.33

- (i) For any recursive $g \geq 2$, $I_P(g\text{-bounded})$ is Π_1^0 complete.
- (ii) For any recursive $g \geq 2$, $I_P(\text{almost } g\text{-bounded})$ is Σ_2^0 complete.
- (iii) $I_P(\text{recursively bounded})$ is Σ_3^0 complete.
- (iv) $I_P(\text{almost recursively bounded})$ is Σ_3^0 complete.
- (v) $I_P(\text{finitely branching})$ is Π_3^0 complete.
- (vi) $I_P(\text{almost bounded})$ is Σ_4^0 complete.

Proof. The upper bounds on the complexity are easily seen by formally writing out the definition. For example, U_e is recursively bounded provided there exists an index a of a total recursive function φ_a such that, for all k and all σ , if $\sigma \smallfrown k \in U_e$, then $k < \varphi_a(\sigma)$. Similarly U_e is almost bounded provided there exists a k such that for all σ of length $\geq k$, there exists an m such that, for all $n > m$, $\sigma \smallfrown n \notin U_e$.

The idea to prove the completeness results is to reduce the appropriate complete set from Theorem 2.32 to the given set. For example, K may be reduced to $I_P(g\text{-bounded})$ by letting

$$U_{h(e)} = \{\sigma : \varphi_{e,t}(e) \uparrow \rightarrow \sigma(t) = 0\}.$$

Then $U_{h(e)} = \{0^t : t \in \omega\}$ and is 2-bounded if $e \notin K$, whereas if $e \in K$ and $\varphi_{e,t}(e) \downarrow$, then $U_{h(e)}$ contains $0^{t-1} \smallfrown m$ for all m and is unbounded.

Similarly, *Fin* may be reduced to $I_P(\text{almost } g\text{-bounded})$, *Rec* may be reduced to $I_P(\text{recursively-bounded})$, and *Coinf* may be reduced to $I_P(\text{finite branching})$. For details, see [26]. $\square$

A subset A of ω is said to be D_n^m if it is the difference of two Σ_n^m sets.

For the remaining theorems, we will omit the results from [26] on almost bounded classes.

Theorem 2.34

(a) For any recursive $g \geq 2$,

$I_P(g\text{-bounded nonempty})$ is Π_1^0 complete,

$I_P(g\text{-bounded empty})$ is D_1^0 complete, and

$(\Pi_1^0, \Sigma_1^0) \leq (I_P(g\text{-bounded nonempty}), I_P(g\text{-bounded empty}))$.

(b) $I_P(r.b. \text{ nonempty})$ is Σ_3^0 complete and $I_P(r.b. \text{ empty})$ is Σ_2^0 complete.

(c) $I_P(\text{bounded nonempty})$ is Π_3^0 complete and $I_P(\text{bounded empty})$ is Σ_2^0 complete.

(d) $(I_P(\text{nonempty}), I_P(\text{empty}))$ is (Σ_1^1, Π_1^1) complete.

Proof. We give the proof of the first part only, where we see our first example of a difference class D_1^0 as well as our first example of double completeness. For the double completeness, we define a reduction h for a given Π_1^0 set A so that $P_{h(e)}$ is always g -bounded and is nonempty if and only if $e \in A$. Let R be a recursive relation so that $e \in A \iff (\forall n)R(e, n)$. Then the map may be defined by putting $0^n \in U_{h(e)} \iff R(e, n)$ and putting no other strings in $U_{h(e)}$.

We see that $I_P(g\text{-bounded empty})$ is D_1^0 as follows. First observe that for a g -bounded tree U_e , $\text{Ext}(U_e)$ has a Π_1^0 characterization,

$$\sigma \in \text{Ext}(U_e) \iff (\forall n > k)(\exists \tau)[|\tau| = n \ \& \ \sigma \prec \tau \ \& \ \tau \in U_e]$$

where the quantifier “ $(\exists \tau)$ ” is bounded by g , since it is equivalent to:

$$(\exists r_1 \leq g(\sigma))(\exists r_2 \leq g(\sigma \smallfrown r_1)) \cdots (\exists r_s \leq g(\sigma \smallfrown (r_1, r_2, \dots, r_{s-1}))).$$

Then we have:

$$e \in I_P(g\text{-bounded empty}) \iff e \in I_P(g\text{-bounded}) \ \& \ \emptyset \notin \text{Ext}(U_e).$$

For the completeness, let $C = B \setminus A$, where A and B are Π_1^0 sets and let R and S be recursive relations so that $e \in A \iff (\forall n) R(e, n)$ and $e \in B \iff (\forall n) S(e, n)$. Then a reduction f of C to $I_P(g\text{-bounded empty})$ is given by putting $\sigma \in U_{f(e)}$ if and only if either

- (i) $(\forall i < |\sigma|) [R(e, i) \ \& \ \sigma(i+1) < g(\sigma \upharpoonright i)]$, or
- (ii) $\sigma = (1 + g(\emptyset) + n)$ where not $S(e, n)$ and $(\forall i < n) (S(e, i))$.

Clearly $U_{f(e)}$ is g -bounded if and only if $e \in B$. Similarly $U_{f(e)}$ is non-empty if and only if $e \in A$. Thus $U_{f(e)}$ is g -bounded and empty iff $e \in B \setminus A$. $\square$

For any cardinal number c and any property $\mathcal{R}$ of trees, let

$$I_P(\mathcal{R} < c) = \{e \in I_P(\mathcal{R}) : \text{card}(P_e) < c\}$$

and similarly define $I_P(\mathcal{R} = c)$ and $I_P(\mathcal{R} > c)$.

Theorem 2.35

- (a) For any positive integer c and any recursive function $g \geq 2$,
 $(I_P(g\text{-bounded} > c), I_P(g\text{-bounded} \leq c))$ is (Σ_2^0, Π_2^0) complete,
 $I_P(g\text{-bounded} = c+1)$ is D_2^0 complete, and
 $I_P(g\text{-bounded} = 1)$ is Π_2^0 complete.
- (b) For any positive integer c , $I_P(r.b. > c)$, $I_P(r.b. \leq c)$ and $I_P(r.b. = c)$ are all Σ_3^0 complete.
- (c) For any positive integer c , $I_P(\text{bounded} \leq c)$ and $I_P(\text{bounded} = 1)$ are both Π_3^0 complete, and $I_P(\text{bounded} > c)$ and $I_P(\text{bounded} = c+1)$ are both D_3^0 complete.
- (d) For any positive integer c , $(I_P(> c), I_P(\leq c))$ is (Σ_1^1, Π_1^1) complete and $I_P(= c)$ is Π_1^1 complete.

Proof. We will just give the proof of the first part. The upper bound on the complexity follows from the fact that a g -bounded Π_1^0 class $P = [T]$ has $> c$ elements if and only if there exist k and incomparable $\sigma_1, \sigma_2, \dots, \sigma_{c+1} \in \omega^k$ such that each $\sigma_i \in \text{Ext}(T)$.

For double completeness, define a reduction of Tot to $I_P(g\text{-bounded} \leq c)$, as follows. For each e , let $\sigma = 0^{m_0} 1^r 0^{m_1} 1^r \dots 0^{m_{k-1}} 1^r 0^{m_k} 1^t \in U_{f(e)}$ if and only if the following conditions are satisfied.

- (1) $1 \leq r \leq c$ and $t \leq r$.
- (2) for each $i < k$, if $\varphi_{e,|\sigma|}(i) \downarrow$, then $\varphi_{e,|\sigma|}(i) = m_i$.
- (3) if $\varphi_{e,|\sigma|}(k) \downarrow$, then $\varphi_{e,|\sigma|}(k) \geq m_k$.

Thus if φ_e is total, then $[U_{f(e)}]$ has exactly c elements,

$$0^{\varphi_e(0)} 1^r 0^{\varphi_e(1)} 1^r \dots$$

for $1 \leq r \leq c$. On the other hand, if φ_e is not total, then $[U_{f(e)}]$ will be infinite.

This reduction also shows the completeness of $I_P(g\text{-bounded} = 1)$.

We need a second reduction j , for an arbitrary Σ_2^0 set B , so that $\text{card}(P_{j(e)}) = 2$ if $e \in B$ and $\text{card}(P_{j(e)}) = 1$ otherwise. For this, let $P_{j(e)}$ be the set of all $(0^{s_0} 1 0^{s_1} 1 \dots)$ such that for all i ,

$$\varphi_{e,s_i+1}(i) \downarrow \ \& \ \varphi_{e,s_i}(i) \uparrow$$

plus the set of all $(0^{s_0} 1 0^{s_1} 1 \dots 0^{s_{k-1}} 1 a^\omega)$ such that $a \leq 1$ and

$$[(\forall i < k) (\varphi_{e,s_i+1}(i) \downarrow \ \& \ \varphi_{e,s_i}(i) \uparrow)] \ \& \ [(\forall s) \varphi_{e,s}(k) \uparrow].$$

Now for the completeness of $I_P(g\text{-bounded} = c + 1)$, let $A = B \cap C$ where B is Σ_2^0 and C is Π_2^0 , let j be the reduction of B just given. Since Tot is Π_2^0 complete, there is a reduction h of C to Tot described above. Then a reduction φ of A to $I_P(g\text{-bounded} = c + 1)$ may be given by defining $U_{\varphi(e)} = U_{j(e)} \oplus U_{f(h(e))}$. $\square$

There are five types of index sets related to finite, countable and uncountable classes of type $\mathcal{R}$:

- (1) $I_P(\mathcal{R} < \aleph_0)$ (finite),
- (2) $I_P(\mathcal{R} \geq \aleph_0)$ (infinite),
- (3) $I_P(\mathcal{R} \leq \aleph_0)$ (countable),
- (4) $I_P(\mathcal{R} = \aleph_0)$ (countably infinite), and
- (5) $I_P(\mathcal{R} > \aleph_0)$ (uncountable).

Theorem 2.36

- (a) For any recursive function $g \geq 2$,
 $(I_P(g\text{-bounded} \geq \aleph_0), I_P(g\text{-bounded} < \aleph_0))$ is (Π_3^0, Σ_3^0) complete.
- (b) $I_P(r.b. \geq \aleph_0)$ is D_3^0 complete and $I_P(r.b. < \aleph_0)$ is Σ_3^0 complete.
- (c) $I_P(\text{bounded} \geq \aleph_0)$ is Π_4^0 complete and $I_P(\text{bounded} < \aleph_0)$ is Σ_4^0 complete.
- (d) $(I_P(\geq \aleph_0), I_P(< \aleph_0))$ is (Σ_1^1, Π_1^1) complete.

Proof. We sketch the proof of the first part. The upper bound on the complexity follows from the uniformity of Theorem 2.35, since

$$e \in I_P(g\text{-bounded} \geq \aleph_0) \iff (\forall c)(e \in I_P(g\text{-bounded} > c)).$$

For the completeness, we define a reduction of Cof to $I_P(g\text{-bounded} < \aleph_0)$ which simultaneously reduces $\omega \setminus Cof$ to $I_P(g\text{-bounded} \geq \aleph_0)$. Let

$$U_{f(e)} = \{0^n : n \in \omega\} \cup \{0^n 1 0^k : n \notin W_{e,k}\}.$$

Then $U_{f(e)}$ is always a binary tree and it is easy to see that

$$P_{f(e)} = \{0^\omega\} \cup \{0^n 1 0^\omega : n \notin W_e\}.$$

Hence $f(e) \in I_P(g\text{-bounded} < \aleph_0) \iff e \in Cof$. □

Theorem 2.37 Let $\mathcal{R}$ be any one of the six notions of boundedness from Theorem 2.33, (including unbounded). Then $(I_P(\mathcal{R} > \aleph_0), I_P(\mathcal{R} \leq \aleph_0))$ is (Σ_1^1, Π_1^1) complete and $I_P(\mathcal{R} = \aleph_0)$ is Π_1^1 complete.

Proof. $I_P(\mathcal{R} > \aleph_0)$ is Σ_1^1 , since for any tree T_e , P_e is uncountable if and only if P_e has a perfect subset, which is if and only if there exists an embedding f from $\{0, 1\}^{<\omega}$ into T_e which preserves the partial order $\prec$. It follows that $I_P(\mathcal{R} \leq \aleph_0)$ and $I_P(\mathcal{R} = \aleph_0)$ are both Π_1^1 .

For the completeness of $I_P(\mathcal{R} > \aleph_0)$, we define a reduction of $I_P(\text{nonempty})$ to $I_P(\text{binary} > \aleph_0)$ as follows. Define the binary tree $U_{f(e)}$ to consist of all strings

$$0^{n_0} \frown \tau_0 \frown 0^{n_1} \frown \tau_1 \frown \dots \frown 0^{n_{k-1}} \frown \tau_{k-1} \frown 0^n,$$

where $(n_0, \dots, n_{k-1}) \in U_e$ and for $i < k$, $\tau_i = (1)$ or $\tau_i = (1, 1)$. Then for any path $x \in [U_e]$, $U_{f(e)}$ will contain uncountably many paths so that if P_e is

nonempty, then $P_{f(e)}$ will be uncountable. If P_e is empty, then every path in $P_{f(e)}$ will end in 0^ω , so that $P_{f(e)}$ will be countable. Note that f also reduces $I_P(\text{empty})$ to $I_P(\text{binary} \leq \aleph_0)$. A reduction g of $I_P(\text{empty})$ to $I_P(\text{binary} = \aleph_0)$ is then given by $U_{g(e)} = U_{f(e)} \cup T$, where T is some primitive recursive tree such that $[T]$ is countably infinite. $\square$

Next, we classify the various index sets of classes with a given recursive cardinality condition. The *recursive cardinality* of a class P is the cardinality of the set of recursive members of P . Also, we say that P is *recursively nonempty* if it has a recursive member and *recursively empty* otherwise.

Theorem 2.38

(a) For any recursive $g \geq 2$,

$(I_P(g\text{-bounded rec. nonempty}), I_P(g\text{-bounded rec. empty}))$ is (Σ_3^0, Π_3^0) complete, and

$I_P(g\text{-bounded nonempty, rec. empty})$ is Π_3^0 complete.

(b) $I_P(r.b. \text{ rec. nonempty})$ is Σ_3^0 complete, and $I_P(r.b. \text{ rec. empty})$, and $I_P(r.b. \text{ nonempty, rec. empty})$ are D_3^0 complete.

(c) $I_P(\text{bounded rec. nonempty})$ is D_3^0 complete, and $I_P(\text{bounded rec. empty})$ and $I_P(\text{bounded nonempty, rec. empty})$ are Π_3^0 complete.

(d) $I_P(\text{rec. nonempty})$ is Σ_3^0 complete, $I_P(\text{rec. empty})$ is Π_3^0 complete, and $I_P(\text{nonempty, rec. empty})$ is Σ_1^1 complete.

Proof. We give the proof for the first part. $I_P(g\text{-bounded rec. nonempty})$ is a Σ_3^0 set, since P_e has a recursive member if and only if

$$(\exists a)[a \in \text{Tot} \ \& \ (\forall n)(\varphi_a \upharpoonright n \in U_e)].$$

It follows that

$$I_P(g\text{-bounded rec. empty}) \quad \text{and} \quad I_P(g\text{-bounded nonempty, rec. empty})$$

are Π_3^0 sets. For the completeness of the first two sets, we define a reduction of Ext_2 to $I_P(g\text{-bounded rec. nonempty})$, by letting $P_{f(a)}$ equal

$$\{x \in \{0, 1\}^\omega : \varphi_a \prec x\} = \{x : (\forall m)(\forall s)(\forall i)[\varphi_{a,s}(m) = i \rightarrow x(m) = i]\}.$$

For the other completeness, let Q be a nonempty, binary Π_1^0 class with no recursive members and let $P_{h(a)} = P_{f(a)} \oplus Q$. $\square$

Theorem 2.39 *Let c be a positive integer.*

(a) *For any recursive $g \geq 2$,*

$(I_P(g\text{-bounded rec.} > c), I_P(g\text{-bounded rec.} \leq c))$ is (Σ_3^0, Π_3^0) complete, and

$I_P(g\text{-bounded rec.} = c)$ is D_3^0 complete.

(b) *$I_P(r.b. \text{ rec.} > c)$ is Σ_3^0 complete and $I_P(r.b. \text{ rec.} \leq c)$ and $I_P(r.b. \text{ rec.} = c)$ are D_3^0 complete.*

(c) *$I_P(\text{bounded rec.} \leq c)$ is Π_3^0 complete and $I_P(\text{bounded rec.} > c)$ and $I_P(\text{bounded rec.} = c)$ are D_3^0 complete.*

(d) *$(I_P(\text{rec.} > c), I_P(\text{rec.} \leq c))$ is (Σ_3^0, Π_3^0) complete and $I_P(\text{rec.} = c)$ is D_3^0 complete.*

Proof. We give the proof of the last part. The upper bound on the complexity follows from the fact that P_e has $> c$ recursive members if and only if there exist $a_0, \dots, a_c \in Tot$ such that

(i) $(\forall i, j \leq c)(\exists m) \varphi_{a_i}(m) \neq \varphi_{a_j}(m)$, and

(ii) $(\forall i \leq c)(\forall n) \varphi_{a_i}[n \in U_e$.

For the completeness of the first two sets, let f be the reduction given in the proof of Theorem 2.38 above for $I_P(g\text{-bounded rec. nonempty})$. Then let Q be a binary Π_1^0 class with exactly $c + 1$ recursive members and let $P_{h(a)} = P_{f(a)} \otimes Q$. For the completeness of the third set, we let $c = 1$ and begin with a construction for unbounded classes using the Σ_3^0 completeness of Rec . For any r.e. set W_a , recall the *modulus function*

$$\mu_a(i) = (\text{least } s) [W_a \cap \{0, 1, \dots, i\} = W_{a,s} \cap \{0, 1, \dots, i\}].$$

It is easy to see that W_a is recursive if and only if μ_a is recursive. Then we define a tree $U_{f(a)}$ so that $P_{f(a)} = \{\mu_a\}$; clearly we then have:

$$f(a) \in I_P(\text{rec.} = 1) \iff a \in Rec.$$

The tree $U_{f(a)}$ is defined so that a string σ of length n is in $U_{f(a)}$ if and only if both:

- (1) $(\forall i < n)(i \in W_{a,n} \iff i \in W_{a,\sigma(i)})$, and
 (2) $(\forall m < n)[\sigma(m) > 0 \rightarrow (\exists i \leq m)(i \in W_{a,\sigma(m)} \setminus W_{a,\sigma(m)-1})]$

Let $A = B \cap C$, where B is Σ_3^0 and C is Π_3^0 . It follows from the completeness of *Rec* and the construction above that there is a recursive function g such that $P_{g(a)}$ is a singleton for each a and has a (unique) recursive member if and only if $a \in B$. Let h be the reduction given above so that $P_{h(a)}$ has exactly c recursive members if $a \in C$ and has at least $c+1$ recursive members if $a \notin C$. Then the function φ defined so that $P_{\varphi(a)} = P_{g(a)} \otimes P_{h(a)}$ is clearly a reduction of $B \cap C$ to $I_P(\text{rec.} = c)$. $\square$

Theorem 2.40 *Let $\mathcal{R}$ be any one of the seven notions of boundedness. Then $(I_P(\mathcal{R} \text{ rec.} < \aleph_0), I_P(\mathcal{R} \text{ rec.} = \aleph_0))$ is (Σ_4^0, Π_4^0) complete.*

Proof. Again, we give the proof for the last part. The characterization of the classes follows from the uniformity in the proof of the previous theorem. For the completeness, let A be a Π_4^0 set so that for some Σ_3^0 relation R ,

$$a \in A \iff (\forall i) R(i, a).$$

As usual, R may be defined so that if $a \notin A$, then $R(i, a)$ for only finitely many values of i . By Theorem 2.38, there is a recursive function f so that for each a and i , $R(i, a)$ if and only if $P_{f(i,a)}$ has a recursive member. Now let $P_{\varphi(a)} = \oplus_i P_{f(i,a)}$. Then it is clear that $a \in A$ if and only if $P_{\varphi(a)}$ has infinitely many recursive members. $\square$

Next we consider the problem of whether a Π_1^0 class has a Δ_2^0 member, that is, a member recursive in $\mathbf{0}'$. The case of recursively bounded classes is omitted, since an r.b. Π_1^0 class has a member recursive in $\mathbf{0}'$ if and only if it is nonempty.

Theorem 2.41

- (a) $(I_P(\Delta_2^0 \text{ nonempty}), I_P(\Delta_2^0 \text{ empty}))$ is (Σ_4^0, Π_4^0) complete, and $I_P(\text{nonempty}, \Delta_2^0 \text{ empty})$ is Σ_1^1 complete.
 (b) $I_P(\text{bounded } \Delta_2^0 \text{ nonempty})$ is Σ_4^0 complete, and $I_P(\text{bounded } \Delta_2^0 \text{ empty})$ and $I_P(\text{bounded nonempty}, \Delta_2^0 \text{ empty})$ are Π_4^0 complete.

Proof. We give the proof of the first part. The upper bounds on the complexity follow from the fact that the set $Tot(\mathbf{0}')$ of indices of total functions recursive in $\mathbf{0}'$ is a Π_3^0 complete set and the characterization

$$e \in I_P(\Delta_2^0 \text{ nonempty}) \iff (\exists a)[a \in Tot(\mathbf{0}') \ \& \ (\forall n)(\varphi_a^{0'}[n \in U_e])].$$

For the completeness of the first two sets, let S be an arbitrary Σ_4^0 set and suppose that

$$a \in S \iff (\exists m)(\forall n)(\exists i)(\forall j) R(a, i, j, m, n)$$

for some recursive relation R . Define the reduction f so that

$$(m, i_0, i_1, \dots) \in P_{f(a)} \iff (\forall n)(\forall j) R(a, i_n, j, m, n).$$

It is clear that if $a \notin S$, then $P_{f(a)}$ is empty and therefore has no member recursive in $\mathbf{0}'$. On the other hand, if $a \in S$, then we may choose m so that

$$(\forall n)(\exists i)(\forall j) R(a, i, j, m, n)$$

and define an infinite path $(m, i_0, i_1, \dots) \in P_{f(a)}$ which is recursive in $\mathbf{0}'$ by letting i_n be the least i such that $(\forall j) R(a, i, j, m, n)$.

For the completeness of the third set, let A be a complete Σ_1^1 set. Since $I_P(\text{nonempty})$ is Σ_1^1 complete, there is a reduction h such that $a \in A$ iff $h(a) \in I_P(\text{nonempty})$. The g reduces A to $I_P(\text{nonempty}, \Delta_2^0 \text{ empty})$ where $P_{g(a)} = P_{f(a)} \oplus Q$ and Q is a Π_1^1 class with no members recursive in $\mathbf{0}'$. $\square$

Many of the results above have parallels for strong Π_2^0 classes. The enumeration of the strong Π_2^0 classes is given by $P_{2,e} = [U_{2,e}]$, where

$$U_{2,e} = \{\sigma : (\forall \tau \preceq \sigma) \tau \in W_e\}.$$

For any property $\mathcal{R}$ of a tree or class, let $I_S(\mathcal{R})$ be the set of indices e such that $U_{2,e}$ has property $\mathcal{R}$. We will say that a tree U is *highly bounded* if it is recursive in $\mathbf{0}'$ and also bounded by a function recursive in $\mathbf{0}'$.

Theorem 2.42

(a) For any recursive $g \geq 2$, $I_S(g\text{-bounded})$ is Π_1^0 complete.

(b) $I_S(r.b.)$ is Σ_3^0 complete.

- (c) $I_S(\text{bounded})$ is Π_3^0 complete.
- (d) For any $g \geq 2$ which is recursive in $\mathbf{0}'$, $I_S(g\text{-bounded})$ is Π_2^0 complete.
- (e) $I_S(\text{highly bounded})$ is Σ_4^0 complete.

Proof. Parts (a) through (c) are proved exactly as in Theorem 2.33 and parts (d) and (e) are simply relativizations of Theorem 2.33. $\square$

Theorem 2.43

- (a) For any recursive $g \geq 2$,
 $(I_S(g\text{-bounded empty}), I_S(g\text{-bounded nonempty}))$ is (Σ_2^0, Π_2^0) complete.
- (b) $I_S(r.b. \text{ nonempty})$ is Σ_3^0 complete, and $I_S(r.b. \text{ empty})$ is Σ_2^0 complete.
- (c) $I_S(\text{bounded nonempty})$ is Π_3^0 complete, and $I_S(\text{bounded empty})$ is Σ_2^0 complete.
- (d) $(I_S(\text{nonempty}), I_S(\text{empty}))$ is (Σ_1^1, Π_1^1) complete.
- (e) For any Δ_2^0 $g \geq 2$,
 $(I_S(g\text{-bounded empty}), I_S(g\text{-bounded nonempty}))$ is (Σ_2^0, Π_2^0) complete.
- (f) $I_S(\text{highly bounded nonempty})$ is Σ_4^0 complete, and
 $I_S(\text{highly bounded empty})$ is Σ_2^0 complete.

Proof. Most of the results either follow from previous results about Π_1^0 classes or are relativizations of those results. We give the proof of the first part.

Observe that if $U_{2,e}$ is finite branching, then the relation “ $\sigma \in \text{Ext}(U_{2,e})$ ” has a Π_2^0 characterization, that is,

$$\sigma \in \text{Ext}(U_{2,e}) \iff (\forall n > |\sigma|)(\exists \tau)[|\tau| = n \ \& \ \sigma \prec \tau \ \& \ \tau \in U_{2,e}].$$

This gives the upper bound on the complexity.

For the completeness, we define a reduction f such that $P_{2,f(e)}$ is always a class of sets and such that $e \in \text{Inf}$ if and only if $P_{2,f(e)}$ is nonempty. Simply let $0^n \in U_{2,f(e)}$ if and only if there exist $a_0 < \dots < a_{n-1}$ each in W_e . $\square$

Theorem 2.44

- (a) For any recursive $g \geq 2$,
 $(I_S(g\text{-bounded rec. nonempty}), I_S(g\text{-bounded rec. empty}))$ is (Σ_3^0, Π_3^0) complete, and
 $I_S(g\text{-bounded nonempty, rec. empty})$ is Π_3^0 complete.
- (b) $I_S(r.b. \text{ rec. nonempty})$ is Σ_3^0 complete, and
 $I_S(r.b. \text{ rec. empty})$ and $I_S(r.b. \text{ nonempty, rec. empty})$ are D_3^0 complete.
- (c) $I_S(\text{bounded rec. nonempty})$ is D_3^0 complete, and
 $I_S(\text{bounded rec. empty})$ and $I_S(\text{bounded nonempty, rec. empty})$ are Π_3^0 complete.
- (d) $(I_S(\text{rec. nonempty}), I_S(\text{rec. empty}))$ is (Σ_3^0, Π_3^0) complete, and
 $I_S(\text{nonempty, rec. empty})$ is Σ_1^1 complete.
- (e) For any $g \geq 2$ and recursive in $\mathbf{0}'$,
 $(I_S(g\text{-bounded rec. nonempty}), I_S(g\text{-bounded rec. empty}))$ is (Σ_3^0, Π_3^0) complete, and
 $I_S(g\text{-bounded nonempty, rec. empty})$ is Π_3^0 complete.
- (f) $I_S(h.b. \text{ rec. nonempty}), I_S(h.b. \text{ rec. empty})$ and
 $I_S(h.b. \text{ nonempty, rec. empty})$ are all Σ_4^0 complete.

Theorem 2.45

- (a) Let $\mathcal{R}$ be any of the notions of boundedness (a)–(c) and (e) from Theorem 2.44. Then,
 $(I_S(\mathcal{R} \Delta_2^0 \text{ nonempty}), I_S(\mathcal{R} \Delta_2^0 \text{ empty}))$ is (Σ_4^0, Π_4^0) complete, and
 $I_S(\mathcal{R} \text{ nonempty}, \Delta_2^0 \text{ empty})$ is Π_4^0 complete.
- (b) $I_S(h.b. \Delta_2^0 \text{ nonempty})$ is Σ_4^0 complete, and
 $I_S(h.b. \Delta_2^0 \text{ empty})$ and $I_S(h.b. \text{ nonempty}, \Delta_2^0 \text{ empty})$ are D_4^0 complete.
- (c) $(I_S(\Delta_2^0 \text{ nonempty}), I_S(\text{rec. empty}))$ is (Σ_4^0, Π_4^0) complete, and
 $I_S(\text{nonempty}, \Delta_2^0 \text{ empty})$ is Σ_1^1 complete.

Recall that a nonempty closed set C is *perfect* if every element of C is a limit point of C , that is, if $D(C) = C$.

Theorem 2.46

- (a) For any recursive function $g \geq 2$, $I_P(g\text{-bounded perfect})$ is Π_3^0 complete.
- (b) $I_P(r.b. \text{ perfect})$ is D_3^0 complete.
- (c) $I_P(\text{bounded perfect})$ is Π_4^0 complete.
- (d) $I_P(\text{perfect})$ is Σ_1^1 complete.

Proof. We give the proof of part (a) only. We have a Π_3^0 condition:

$e \in I_P(g\text{-bounded perfect})$ if and only if

$$(\forall \sigma)[\sigma \in \text{Ext}(U_e) \rightarrow (\exists \tau)(\exists i \neq j)(\sigma \prec \tau \ \& \ \tau \frown i, \tau \frown j \in \text{Ext}(U_e))].$$

For the completeness, modify the proof of Theorem 2.36 by letting $U_{h(e)}$ contain $\{0^n : n \in \omega\}$ together with all strings $0^n 1 \frown \sigma_1 \dots \sigma_k$ where $n \notin W_{e,k}$ and each σ_i is either (010) or (011) . $\square$

Recall that a set is *meager* if its the countable union of nowhere dense sets. It is easy to see that a closed set P is meager if and only if it contains no open interval. P is *meager in* Q for a set Q if $P \cap Q$ is meager.

Theorem 2.47 $I_P(\text{meager})$ and $I_P(\text{meager in } \{0,1\}^\omega)$ are both Π_2^0 complete.

Proof. P_e is non-meager if and only if, for some σ , $I(\sigma) \subseteq P_e$, that is, if and only if

$$(\exists \sigma)(\forall \tau)[\sigma \prec \tau \rightarrow \tau \in U_e].$$

For the completeness, let A be a Σ_2^0 set and let R be a recursive relation so that $a \in A \iff (\exists m)(\forall n) R(m, n, a)$. Then a reduction of A to $I_P(\text{non-meager})$ is given by

$$U_{f(a)} = \{m \frown \tau : \tau \ \& \ (\forall n < |\tau|) R(m, n, a)\}.$$

For the binary version, let

$$U_{g(a)} = \{0^m : m \in \omega\} \cup \{(0^m) \frown 1 \frown \tau : \tau \in \{0,1\}^{<\omega} \ \& \ (\forall n < |\tau|) R(m, n, a)\}.$$

$\square$

Let λ be the standard product measure on $\{0, 1\}^\omega$ which gives $\lambda(I(\sigma)) = 2^{-|\sigma|}$. Let $I_P(\text{measure} \leq r) = \{e : \lambda(P_e \cap \{0, 1\}^\omega) \leq r\}$, and similarly for equality and the other inequalities. The following result is presented without proof. See [26] for details.

Theorem 2.48

- (a) For any Σ_1^0 real $r \in (0, 1]$, $(I_P(\text{measure} < r), I_P(\text{measure} \geq r))$ is (Σ_1^0, Π_1^0) complete (so that $I_P(\text{measure} = 1)$ is Π_1^0 complete) and, if r is not recursive, then $I_P(\text{measure} \leq r)$ is Σ_1^0 complete.
- (b) For any Π_1^0 real $r < 1$, $(I_P(\text{measure} > r), I_P(\text{measure} \leq r))$ is (Σ_2^0, Π_2^0) complete and $I_P(\text{measure} = r)$ is Π_2^0 complete; if r is Π_1^0 complete, then $(I_P(\text{measure} < r), I_P(\text{measure} \geq r))$ is (Σ_2^0, Π_2^0) complete.

Next we consider the problem of separating sets. For any two sets A and B of natural numbers, let $S(A, B)$ contain those sets C such that $A \subseteq C$ and $B \cap C = \emptyset$. Thus $S(\emptyset, B) = \mathcal{P}(B)$ and $S(A, \emptyset)$ is the family of supersets of A . Of course for any A and B , $S(A, B)$ is either finite (if the symmetric difference of A and B is finite) or has cardinality of the continuum. For any property $\mathcal{R}$, let

$$SS(\mathcal{R}) = \{[a, b] : S(W_a, W_b) \text{ has property } \mathcal{R}\}$$

Theorem 2.49

- (a) $(SS(\text{empty}), SS(\text{nonempty}))$ is (Σ_1^0, Π_1^0) complete.
- (b) For any positive integer c , $(SS(> 2^c), SS(\leq 2^c))$ is (Σ_2^0, Π_2^0) complete, $SS(= 2^c)$ is D_2^0 complete, and $SS(= 1)$ is Π_2^0 complete.
- (c) $(SS(\text{finite}), SS(\text{infinite}))$ is (Σ_3^0, Π_3^0) complete.
- (d) $(SS(\text{rec. nonempty}), SS(\text{rec. empty}))$ is (Σ_3^0, Π_3^0) complete, and $SS(\text{nonempty, rec. empty})$ is Π_3^0 complete.

Proof. Note that there is a primitive recursive function ψ such that $S(W_a, W_b) = P_{\psi(a,b)}$ for each a and b . The upper bound on the complexities now follow from the corresponding upper bounds on the complexities of Π_1^0 classes.

We sketch the proofs of the completeness. Fix b and c such that $W_b = \emptyset$ and $W_c = \omega$.

(a) This follows from the fact that $S(W_e, W_c) = \emptyset \iff W_e \neq \emptyset$.

(b) Observe that $|S(W_e, W_b)| = 2^c$ if and only if $|\omega \setminus W_e| = c$. Now $e \in Tot \iff \omega \setminus W_e = \emptyset$. Hence

$$e \in Tot \iff [e, b] \in SS(= 1) \iff [e, b] \in SS(\leq 1).$$

This also proves the completeness of $SS(> 1)$ as well. If we let

$$W_{\varphi(e)} = \{x + c : x \in W_e\},$$

then $|\omega \setminus W_e| = c + |\omega \setminus W_e|$. Thus $e \in Tot \iff [\varphi(e), b] \in SS(\leq 2^c)$. The completeness result for $SS(> 2^c)$ now immediately follows.

Next we consider the D_2^0 completeness of $SS(= 2^c)$. It follows from the Π_2^0 -completeness of Tot and our remarks above that for a given Π_2^0 set A , there is a recursive function f such that if $a \in A$, then $|\omega \setminus W_a| = c$ and if $a \notin A$, then $|\omega \setminus W_a| > c$. Let B be a Σ_2^0 set. We will obtain a reduction g such that if $a \in B$, then $|\omega \setminus W_e| = 0$ and if $a \notin B$, then $|\omega \setminus W_a| = 1$. We define a reduction for the Σ_2^0 complete set Fin . Given an index e , construct the r.e. set $W_{g(e)}$ in stages, so that after stage s , we have the set $W_{\varphi(e),s}$ and also a number x_s which is intended to be the unique member of $\omega \setminus W_e$, if any. We assume as usual that at most one element comes into W_e at any stage s . The construction begins with $W_{\varphi(e),0} = \emptyset$ and $x_0 = 0$. At stage $s + 1$, there are two cases.

CASE 1: If no element comes into W_e , or if an element $x < x_s$ comes into W_e , then we let $x_{s+1} = x_s$ and we put $s + 1 \in W_{\varphi(e),s+1}$. In this case, $W_{\varphi(e),s+1} = \{0, 1, \dots, s + 1\} \setminus \{x_s\}$.

CASE 2: If an element $x \geq x_s$ comes into W_e , then we put $x_s \in W_{\varphi(e),s+1}$ and let $x_{s+1} = s + 1$; in this case $W_{\varphi(e),s+1} = \{0, 1, \dots, s\}$.

If W_e is finite, then at some stage, we obtain x_s greater than every element of W_e , so that Case 1 applies at every later stage t . Thus $x_t = x_s$ for all $t > s$ and $\omega \setminus W_{\varphi(e)} = \{x_s\}$. If W_e is infinite, then Case 2 applies infinitely often and $W_{\varphi(e)} = \omega$. Finally, we define a reduction of the D_2^0 set $A \cap B$ to $SS(= c + 1)$ by letting $W_{h(e)} = W_{f(e)} \oplus W_{g(e)}$.

(c) Observe that $S(W_a, W_b)$ is finite if and only if W_a is cofinite and apply Theorem 2.32.

(d) Recalling the proof of Theorem 2.38, we have a reduction f of a Π_3^0 complete set Ext_2 so that

$$P_{f(e)} = \{x \in \{0, 1\}^\omega : \varphi_e \prec x\} = S(W_{g(e)}, W_{h(e)}),$$

where $W_{g(e)} = \{n : \varphi_e(n) = 1\}$ and $W_{h(e)} = \{n : \varphi_e(n) = 0\}$. Thus

$$\begin{aligned} e \in Ext_2 &\iff [h(e), g(e)] \in SS(\text{rec. nonempty}) \\ &\iff [h(e), g(e)] \in SS(\text{nonempty, rec. nonempty}). \quad \square \end{aligned}$$

We end this section with a remark about the index sets for the set of k -ary trees since the set of solutions of many of the recursive mathematical problems we consider later can be naturally coded as the set of paths through a k -ary tree. We note that it is easy to see that there is an effective list of all primitive recursive k -ary trees, $T_0^k, T_1^k, \dots$. Then for any property $\mathcal{R}$ of k -ary trees, we can form the index set

$$I_{P,k}(\mathcal{R}) = \{e : T_e^k \text{ has property } \mathcal{R}\}.$$

Almost all the results about g -bounded trees where $g(x) = k$ for all x hold for the index sets $I_{P,k}(\mathcal{R})$. That is, $I_P(g\text{-bounded})$ is only Π_1^0 complete. Thus if we compare the index sets $I_{P,k}(\mathcal{R})$ and the index set $I_P(g\text{-bounded} \ \& \ \mathcal{R})$, the extra $\forall$ quantifier is absorbed for index sets whose complexity is in the second or higher level of the arithmetic hierarchy.

Theorem 2.50

- (i) $I_{P,k}(\text{empty})$ is Σ_1^0 complete, and $I_{P,k}(\text{nonempty})$ is Π_1^0 complete.
- (ii) $I_{P,k}(\text{dec})$ is Π_1^0 complete.
- (iii) For any positive integer c , $(I_{P,k}(> c), I_{P,k}(\leq c))$ is (Σ_2^0, Π_2^0) complete, $I_{P,k}(= c + 1)$ is D_2^0 complete, and $I_{P,k}(= 1)$ is Π_2^0 complete.
- (iv) $(I_{P,k}(\geq \aleph_0), I_{P,k}(< \aleph_0))$ is (Σ_3^0, Σ_3^0) complete.
- (v) $(I_{P,k}(\text{rec. nonempty}), I_{P,k}(\text{rec. empty}))$ is (Σ_3^0, Π_3^0) complete.
- (vi) For any positive integer c , $(I_{P,k}(\text{rec. } > c), I_{P,k}(\text{rec. } \leq c))$ is (Σ_3^0, Π_3^0) complete, and $I_{P,k}(\text{rec. } = c)$ is D_3^0 complete.

(vii) $(I_{P,k}(\text{rec.} < \aleph_0), I_{P,k}(\text{rec.} = \aleph_0))$ is (Σ_4^0, Π_4^0) complete.

(viii) $I_{P,k}(\text{perfect})$ is Π_3^0 complete.

Proof.

(i) $I_{P,k}(\text{empty})$ is Σ_1^0 since by König's Lemma,

$$[T_e^k] = \emptyset \iff (\exists n)(\forall \sigma \text{ with } |\sigma| = n)(\sigma \notin T).$$

It thus follows that $I_{P,k}(\text{nonempty})$ is Π_1^0 .

To see that $I_{P,k}(\text{empty})$ is Σ_1^0 complete, we define a recursive function f which reduces $I_{P,k}(\text{empty})$ to K by putting $\emptyset \in T_{f(e)}^k$ and for all other $\sigma \in \{0, \dots, k-1\}^*$, letting

$$\sigma \in T_{f(e)}^k \iff e \notin K^{|\sigma|}.$$

Then it is clear that $f(e) \in I_{P,k}(\text{empty})$ iff $e \in K$. Clearly f also shows that $I_{P,k}(\text{nonempty})$ is Π_1^0 complete.

(ii) $I_{P,k}(\text{dec})$ is Π_1^0 since T_e^k is decidable iff

$$(\forall \sigma \in T_e^k)(\exists i < k)(\sigma \frown i \in T_e^k).$$

The function f of part (a) shows that $I_{P,k}(\text{dec})$ is Π_1^0 complete.

(iii)–(viii) For any of the properties $\mathcal{R}$ in parts (iii)–(viii), simply use the same proofs that were used to prove the complexity of $I_p(g\text{-bounded} \ \& \ \mathcal{R})$. $\square$

3 Logical Theories

In this section, we look at the interaction between Π_1^0 classes and logical theories. The first problem that we will consider is the problem of finding a complete consistent extension of a first-order logical theory. A recursively presented instance of a logical theory will be an axiomatizable theory, that is a theory with a recursively enumerable set of axioms. We will show that the set of complete consistent extensions of an axiomatizable theory can always be represented by an r.b. Π_1^0 class and, vice versa, that any Π_1^0 class can be represented by the set of complete consistent extensions of some first order logical theory. We will also consider the related, but simpler, problem of finding a consistent extension of a first-order logical theory.

We begin with some definitions and background on logical theories. Recall the arbitrary first-order effective language $\mathcal{L}$ described in the Introduction. Let $Sent(\mathcal{L})$ be the set of sentences of $\mathcal{L}$. For any subset Γ of $Sent(\mathcal{L})$, the set $Con(\Gamma)$ of consequences of Γ is the closure of Γ under logical deduction and the set $Ref(\Gamma)$ of refutations of Γ is the set of negations of the consequences of Γ . A subset Γ of $Sent(\mathcal{L})$ is a first order logical theory if Γ is closed under logical deduction. Σ is said to be a set of axioms for Γ if $\Gamma = Con(\Sigma)$ and Γ is *axiomatizable* if Γ has a recursive set of axioms. It is not hard to see that Γ is axiomatizable if and only if Γ is recursively enumerable. A theory is said to be *decidable* if it is recursive. It follows from Post's Theorem that a complete axiomatizable theory is decidable. The classical result here is that any consistent theory has an extension to a complete consistent theory and follows as usual from Zorn's Lemma.

It is possible to trace the study of Π_1^0 classes back to the discovery of the undecidability of arithmetic by Turing and Church (independently) in 1936, following soon after Gödel's incompleteness theorem. The problem associated with a logical theory is to find a complete consistent extension of that theory and the associated class is the family of all such extensions. The basic undecidability result is that there is no decidable, complete, consistent extension of Peano arithmetic. The set of all complete consistent extensions of Peano arithmetic may be viewed as a Π_1^0 class and hence provides an example of a nonempty Π_1^0 class with no recursive member. This led to the definition of an *essentially undecidable theory* as a theory with no decidable complete consistent extension.

Now if Σ is any consistent complete extension of a theory Γ , then Σ separates the set T of consequences of Γ from the set R of refutations of Γ . A theory is said to be *separable* if the consequences and refutations can be separated by a recursive set and is otherwise said to be *inseparable*. Rosser [133] observed also in 1936 that Peano arithmetic is an inseparable theory and that any inseparable theory is essentially undecidable. This also provided the first example of recursively inseparable r.e. sets. Ehrenfeucht showed in 1961 [47] that there are separable theories which are essential undecidable. His construction, using theories of propositional calculus, also shows that every Π_1^0 class may be represented as the set of complete consistent extensions of a theory.

A complete, consistent extension of Peano Arithmetic is of course just the theory of some (possibly non-standard) model of Peano arithmetic.

The theory of Peano arithmetic is of great interest in mathematical logic, due in part to the connection with Gödel's Incompleteness Theorem, and has been developed in the papers of Jockusch and Soare [75, 74], Knight [87], Marker [108] and many others. Shoenfield observed in [137] that in general, the family of complete, consistent extensions of an axiomatizable first order theory can be represented by a Π_1^0 class.

The notion of a thin Π_1^0 class arose from the study of logical theories. Martin and Pour-El [109] constructed an axiomatizable, essentially undecidable theory T such that each axiomatizable extension of T is a finite extension of T . The collection P of complete, consistent extensions of this theory T is the first known example of a thin Π_1^0 class. Here an axiomatizable extension S of T is an r.e. set and the class Q of complete, consistent, axiomatizable extensions of S is a Π_1^0 subclass of P . In general, if the set P of complete consistent extensions of a theory T is a thin Π_1^0 class, then each axiomatizable extension of T is a finite extension. The relation between thin Π_1^0 classes and axiomatizable theories was first developed in Downey's thesis [35], which contains some of the results of this section. The reader is referred to Odifreddi [121, pp. 349–360] for further discussion of axiomatizable theories.

In the following, the expression $\bigwedge\{\gamma : \gamma \in S\}$ denotes the conjunction of the finite set of sentences S .

Theorem 3.1 *For any r.e. theory Γ of an effective language $\mathcal{L}$, both the class of consistent extensions of Γ and the class of complete consistent extensions of Γ can be represented as Π_1^0 classes. Furthermore, if Γ is a decidable theory, then these classes can be represented by recursive trees with no dead ends.*

Proof. Let $\mathcal{L}$ be an effective first-order language and let $S = \text{Sent}(\mathcal{L})$ have an effective enumeration as $\gamma_0, \gamma_1, \dots$. Then the sentence γ_i may be identified with its index i , so that a theory Γ is represented by the set $\{i : \gamma_i \in \Gamma\}$, and a class of theories is represented by a class in $\{0, 1\}^\omega$. Let $\Gamma \vdash_s \gamma_i$ be the recursive relation of s and i which means that there is a proof of γ_i from Γ of length s . Then the class $P(\Gamma)$ of complete consistent extensions of Γ may be represented by the set of infinite paths through the recursive tree T defined so that for any

$$\sigma = (\sigma(0), \dots, \sigma(n-1)),$$

σ is in T if and only if the following conditions hold.

- (1) For any $i < n$, if $\Gamma \vdash_n \gamma_i$, then $\sigma(i) = 1$.
- (2) For any $i, j < n$, if $\Gamma \vdash_n \gamma_i \rightarrow \gamma_j$ and $\sigma(i) = 1$, then $\sigma(j) = 1$.
- (3) For any $i, j, k < n$, if $\gamma_k = (\gamma_i \ \& \ \gamma_j)$, $\sigma(i) = 1$ and $\sigma(j) = 1$, then $\sigma(k) = 1$.
- (4) For any $i, j < n$, if $\sigma(i) = 1$ and $\gamma_j = \neg\gamma_i$, then $\sigma(j) = 0$.
- (5) For any $i, j < n$, if $\gamma_j = \neg\gamma_i$, then either $\sigma(i) = 1$ or $\sigma(j) = 1$.

Let x be an infinite path through T and let $\Delta = \{\gamma_i : x(i) = 1\}$. Condition (1) ensures that $\Gamma \subseteq \Delta$, while conditions (1), (2), and (3) ensure that Δ is a theory. Condition (4) ensures that Δ is consistent and condition (5) ensures that Δ is complete. To represent the class of consistent extensions of Γ , simply omit the final clause (5).

If Γ is decidable, then in each case we can modify the clauses given above as follows to get a tree S with no dead ends which has the same class of infinite paths. First, combine the first three clauses into the statement:

- (1') For any $k < n$, if $\Gamma \vdash \wedge\{\gamma_i : i < n \ \& \ \sigma(i) = 1\} \rightarrow \gamma_k$, then $\sigma(k) = 1$.

Next, replace clause (4) with

- (4') It is not the case that $\Gamma \vdash [\wedge\{\gamma_i : i < n \ \& \ \sigma(i) = 1\} \rightarrow (\gamma_0 \ \& \ \neg\gamma_0)]$.

It follows that for any $\sigma \in S$,

$$\Gamma \cup \{\gamma_i : i < |\sigma| \ \& \ \sigma(i) = 1\} \cup \{\neg\gamma_i : i < n \ \& \ \sigma(i) = 0\}$$

is consistent and therefore has an extension to a complete consistent theory $\Gamma(\sigma)$ which will be represented by an extension of σ . Thus S has no dead ends. $\square$

We note that a recursively enumerable Boolean algebra $L(\Gamma)$ (the *Lindenbaum algebra*) may be associated with the theory Γ and the class of complete consistent extensions of Γ may be viewed as the class of maximal filters of $L(\Gamma)$. This connection will be discussed further in Section 5.

We can now derive a number of immediate corollaries from the results of Section 2. Parts (i) through (v) of (a) follows from Theorems 3.6, 3.7, 3.14, 3.12 and 3.10, respectively.

Theorem 3.2

- (a) *For any consistent, axiomatizable first-order theory Γ :*
- (i) *Γ has a complete consistent extension in some r.e. degree.*
 - (ii) *Γ has complete consistent extensions Γ_1 and Γ_2 such that any function recursive in both Γ_1 and Γ_2 is recursive.*
 - (iii) *If Γ has only countably many complete consistent extensions, then Γ has a decidable complete consistent extension.*
 - (iv) *If Γ has only finitely many complete consistent extensions, then every complete consistent extension is decidable.*
 - (v) *If Γ has no decidable complete consistent extensions, then for any countable sequence of nonzero degrees $\{\mathbf{a}_i\}$, Γ includes a continuum of complete consistent extensions Σ which are pairwise Turing incomparable and such that the degree of Σ is incomparable with each $\mathbf{a}_i$.*
- (b) *Any consistent, decidable first order theory Γ has a complete, consistent, decidable extension.*

Next we turn to the other direction of our correspondence, that is, representing an arbitrary Π_1^0 class by the set of solutions to one of our two problems.

Theorem 3.3 *Any r.b. Π_1^0 class P can be represented by the set of complete, consistent extensions of an axiomatizable theory. Furthermore, this theory may be taken to be propositional and it also may be taken from the language with a single binary relation.*

Proof. We first give the proof due to Ehrenfeucht [47]. Let the language $\mathcal{L}$ consist of a countable sequence $A_0, A_1, \dots$ of 0-place relations symbols, that is, propositional variables. For any $x \in \{0, 1\}^\omega$, we can define a complete consistent theory $\Delta(x)$ for $\mathcal{L}$ to be $\text{Con}(\{C_i : i \in \omega\})$, where $C_i = A_i$ if $x(i) = 1$ and $C_i = \neg A_i$ if $x(i) = 0$. It is clear that every complete consistent theory of $\mathcal{L}$ is one of these. Thus for any Π_1^0 class $P \subseteq \{0, 1\}^\omega$, we let want a theory Γ such that $\Delta(P) = \{\Delta(x) : x \in P\}$ is the set of complete, consistent extensions of Γ . For each finite sequence $\sigma = (\sigma(0), \dots, \sigma(n-1))$, let $P_\sigma = C_0 \wedge C_1 \wedge \dots \wedge C_{n-1}$, where $C_i = A_i$ if $\sigma(i) = 1$ and $C_i = \neg A_i$ if $\sigma(i) = 0$.

Let the binary tree T be given such that $P = [T]$ and define the theory $\Gamma(T)$ to consist of all $P_\sigma \rightarrow A_n$ such that $\sigma \in T$ and $\sigma \frown 0 \notin T$ and all $P_\sigma \rightarrow \neg A_n$ such that $\sigma \in T$ and $\sigma \frown 1 \notin T$, where $|\sigma| = n$. We claim that $\Delta(P)$ is in fact equal to the set of complete consistent extensions of $\Gamma(T)$. Suppose first that $x \in P$ and let $\text{Con}(\{C_i : i \in \omega\}) = \Delta(x)$. Now any $\gamma \in \Gamma(T)$ is of the form $P_\sigma \rightarrow \pm A_i$ for some $\sigma \in T$; say that $|\sigma| = n$. There are several cases. If $\sigma \neq x \frown n$, then $\Delta(x) \vdash \neg P_\sigma$, so that we always have $\Delta(x) \vdash P_\sigma \rightarrow \pm A_n$. Thus we may suppose that $\sigma = x \frown n$. If $\sigma \frown 0 \notin T$, then of course $x(n) = 1$, so that $C_n = A_n \in \Delta(x)$ and therefore $\Delta(x) \vdash P_\sigma \rightarrow A_n$. Similarly, if $\sigma \frown 1 \notin T$, then $\Delta(x) \vdash P_\sigma \rightarrow \neg A_n$. Thus $\Delta(x)$ is a complete consistent extension of $\Gamma(T)$. On the other hand, let Δ be a complete consistent extension of $\Gamma(T)$. Then, for each i , we have either $\Delta \vdash A_i$ or $\Delta \vdash \neg A_i$; let $C_i = A_i$ if $A_i \in \Delta$ and $C_i = \neg A_i$ otherwise. Define $x \in \{0, 1\}^\omega$ so that $x(i) = 1$ if and only if $\Delta \vdash A_i$. Then clearly $\Delta = \Delta(x)$. It remains to be shown that $x \in P$. Now if $x \notin P$, then there is some n such that $\sigma = x \frown n + 1 \notin T$ and $x \frown n \in T$. Then $P_\sigma = C_0 \wedge \cdots \wedge C_{n-1}$, so that $\Delta \vdash P_\sigma$, and $P_\sigma \rightarrow \neg C_i \in \Gamma(T)$, so that Δ is not consistent with $\Gamma(T)$. This contradiction proves that $\Delta = \Delta(x)$.

This proof was modified by Jockusch and Soare in [75, p. 54] for a language with one binary relation R . The underlying axioms assert that R is an equivalence relation and that, for any n , there are either one or two equivalence classes consisting of exactly n members. The propositional statement A_n in the proof above is replaced by the statement that there is exactly one equivalence class with n elements. $\square$

The representation Theorem 3.3 has a number of immediate corollaries.

First we briefly discuss the significance of isolated and limit points and of thin and minimal Π_1^0 classes in this context. Let Γ be an axiomatizable theory and let Σ be a theory which is a consistent axiomatizable extension of Γ . By abuse of notation, let us say that the set of complete consistent extensions of Γ is a Π_1^0 class, denoted by $P(\Gamma)$ and likewise for Σ . Then $P(\Sigma)$ is a Π_1^0 subclass of $P(\Gamma)$. Thus if $P(\Gamma)$ is thin, then $P(\Sigma)$ is a relatively clopen subclass of $P(\Gamma)$, so that Σ is a finitely generated extension of Γ . Let us explain this further. Given that $P(\Sigma)$ is a relatively clopen subclass of $P(\Gamma)$, let $U = I(\sigma_1) \cup I(\sigma_2) \cup \cdots \cup I(\sigma_k)$ be a clopen set such that $P(\Sigma) = P(\Gamma) \cap U$. For each i , let $R_i = P_{\sigma_i}$, as defined above. Let $A = R_1 \vee R_2 \vee \cdots \vee R_k$. Then it is clear that, for any Δ ,

$$\Delta \in P(\Sigma) \text{ if and only if } \Delta \in P(\Gamma) \text{ and } A \in \Delta. \quad (*)$$

It follows that Σ is generated by $\Gamma \cup \{A\}$. Since any consistent set of sentences can be extended to a complete consistent theory, we see that for any sentence γ and any set Λ of sentences γ , $\Lambda \vdash \gamma$ if and only if $\gamma \in \Delta$ for every complete consistent extension Δ of Λ . Clearly if $\Gamma \cup \{A\} \vdash \gamma$, then $\Sigma \vdash \gamma$, since $A \in \Sigma$. On the other hand, if $\Sigma \vdash \gamma$, then $\gamma \in \Delta$ for every $\Delta \in P(\Sigma)$. It follows from (*) that $\gamma \in \Delta$ for every $\Delta \in P(\Gamma \cup \{A\})$, so that $\Gamma \cup \{A\} \vdash \gamma$. An axiomatizable theory Γ with this property (that every consistent axiomatizable extension of Γ is finitely generated) is said to be a *Martin–Pour-El theory*. This generalizes the original notion of Martin and Pour-El [109], where such theories are assumed to be essentially undecidable. It was shown by Downey, Jockusch and Stob [41] that the set of r.e. degrees containing essentially undecidable Martin–Pour-El theories is precisely the a.n.r. degrees.

If $P(\Gamma)$ is minimal, then either $P(\Sigma)$ is finite or else $P(\Gamma) \setminus P(\Sigma)$ is finite. If the complete consistent extension Δ of Γ is isolated in $P(\Gamma)$, then $\{\Delta\}$ is a relatively clopen subset of $P(\Gamma)$, so that Δ is a finitely generated extension of Γ as above.

Parts (i) through (viii) of the next theorem follow from Theorems 3.8, 3.9, 3.16 and 3.20.

Theorem 3.4

- (i) *There is a consistent axiomatizable first-order theory Γ which has no recursive consistent complete extension.*
- (ii) *There is a consistent axiomatizable first-order theory Γ such that any two distinct complete consistent extensions of Γ are Turing incomparable, where distinct means having infinite symmetric difference (in the Lindenbaum algebra).*
- (iii) *If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} <_T \mathbf{0}'$, then there exists a consistent axiomatizable first-order theory Γ such that Γ has no complete consistent extension of degree $\mathbf{a}$ and has no decidable complete consistent extension.*
- (iv) *There is a consistent axiomatizable first-order theory Γ such that if $\mathbf{a}$ is the degree of any complete consistent extension of Γ and $\mathbf{b}$ is an r.e. degree with $\mathbf{a} <_T \mathbf{b}$, then $\mathbf{b} \equiv_T \mathbf{0}'$.*

- (v) If $\mathbf{c}$ is any r.e. degree, then there exists a consistent axiomatizable first-order theory Γ such that the set of r.e. degrees which contain complete consistent extensions of Γ equals the set of r.e. degrees $\geq_T \mathbf{c}$.
- (vi) There is a consistent axiomatizable first-order theory Γ such that if Δ is a complete consistent extension of Γ with $\Delta <_T \mathbf{0}'$, then there exists a nonrecursive r.e. set A such that $A <_T \Delta$.
- (vii) For any degree $\mathbf{a}$ such that either $\mathbf{a} \leq_T \mathbf{0}'$ or $\mathbf{0}' \leq_T \mathbf{a} \leq_T \mathbf{0}''$ such that $\mathbf{a}$ is comparable with $\mathbf{0}'$, there exists a consistent first-order axiomatizable theory Γ with a complete consistent extension Δ of degree $\mathbf{a}$ such that Δ is the unique undecidable complete consistent extension of Γ and such that any other complete consistent extension of Γ is finitely generated.
- (viii) There is a Martin-Pour-El theory Γ with a complete consistent extension Δ of degree $\mathbf{0}'$ such that Δ is the unique undecidable complete consistent extension of Γ , such that any other complete consistent extension of Γ is finitely generated, and such that for any axiomatizable extension Σ of Γ , either there are only finitely many complete consistent extensions of Σ or else all but finitely many complete consistent extensions of Γ extend Σ .

We next consider the special case of Π_1^0 classes with no dead ends.

Theorem 3.5 *Let T be a highly recursive tree with no dead ends and let $P = [T]$. Then there is a decidable theory Γ and an effective one-to-one correspondence between the complete consistent extensions of Γ and P .*

Proof. This follows from the proof of Theorem 3.3. We claim that if the tree T has no dead ends, then the resulting theory $\Gamma(T)$ is a decidable theory.

Let a sentence $\gamma = \gamma(A_0, \dots, A_{n-1})$ of the language $\mathcal{L}$ be given. We claim that $\Gamma(T) \vdash \gamma$ if and only if $\bigwedge \{P_\sigma \vdash \gamma : \sigma \in T \text{ \& } |\sigma| = n\}$, that is, if and only if $P_\sigma \vdash \gamma$ for all $\sigma \in T$ with $|\sigma| = n$. This claim clearly implies that $\Gamma(T)$ is decidable.

Recall first from the proof of Theorem 3.3 that $\Gamma(T) \vdash \gamma$ if and only if $\Delta(x) \vdash \gamma$ for every $x \in [T]$.

We argue by the contrapositive. Suppose first that $\Gamma(T) \not\vdash \gamma$. Then there is some $x \in [T]$ such that $\Delta(x) \vdash \neg\gamma$. Since γ only depends on $A_0, \dots, A_{n-1}$, it follows that $P_\tau \vdash \neg\gamma$, where $\tau = x \upharpoonright n \in T$. Thus $P_\tau \vdash \gamma$ is clearly false,

making it also false that $\bigwedge \{P_\sigma \vdash \gamma : \sigma \in T \ \& \ |\sigma| = n\}$. Suppose next that $\bigwedge \{P_\sigma \vdash \gamma : \sigma \in T \ \& \ |\sigma| = n\}$ is false. Then $P_\tau \vdash \gamma$ is false for some fixed $\tau \in T$, which means that $P_\tau \vdash \neg\gamma$ (since γ depends only on $A_0, \dots, A_{n-1}$). Since T has no dead ends, there is some $x \in P$ such that $\tau \prec x$ and therefore $\Delta(x) \vdash \neg\gamma$. It follows from the remark above that $\Gamma(T) \not\vdash \gamma$. $\square$

We can now apply Theorem 3.20 (a) to obtain the following.

Theorem 3.6 *There is a decidable Martin-Pour-El theory Γ with a complete consistent extension Δ such that Δ is the unique undecidable complete consistent extension of Γ , such that any other complete consistent extension of Γ is finitely generated, and such that for any axiomatizable extension Σ of Γ , either there are only finitely many complete consistent extensions of Σ or else all but finitely many complete consistent extensions of Γ extend Σ .*

We will now consider the notion of index sets for axiomatizable theories. Since an axiomatizable theory $\Gamma \subseteq \{\gamma_0, \gamma_1, \dots\}$ may be identified with the recursively enumerable set $\{i : \gamma_i \in \Gamma\}$, we will let $\Gamma_e = \{\gamma_i : i \in W_e\}$ be the e -th set of axioms. Thus, whenever Γ_e is closed under implication, Γ_e will be the e -th axiomatizable theory.

We will apply the results of Section 2.7 on index sets of Π_1^0 classes.

Theorem 3.7

- (a) $Cons = \{e : \Gamma_e \text{ is consistent}\}$
 $= \{e : \Gamma_e \text{ has a complete consistent extension}\}$ is a Π_1^0 complete set.
- (b) $Theor = \{e : \Gamma_e \text{ is a theory}\}$ is a Π_2^0 complete set.
- (c) $Theor_1 = \{e \in Theor : \Gamma_e \text{ is a consistent theory}\}$ is a Π_2^0 complete set.
- (d) $Cmpl = \{e : \Gamma_e \text{ is a complete consistent theory}\}$ is a Π_2^0 complete set.
- (e) $\{e \in Theor_1 : \Gamma_e \text{ has a decidable complete consistent extension}\}$ is a Σ_3^0 complete set.

Proof.

- (a) $Cons$ is a Π_1^0 set, since

$$e \in Cons \iff (\forall s) \neg [\bigwedge \{\gamma_i : i \in W_{e,s}\} \rightarrow (\gamma_0 \wedge \neg\gamma_0)].$$

For the completeness, we define a reduction of $\omega \setminus K$ to *Cons* by choosing a fixed propositional sentence A (or in general any sentence such that $\neg A$ is consistent by itself) and defining

$$\Gamma_{f(a)} = \{\neg A\} \cup \{\gamma_e : a \in W_a \text{ \& } (\exists n < e)(\gamma_e = \bigwedge_{i=1}^n A)\}.$$

Then if $a \notin K$, we have $\Gamma_{f(a)} = \{\neg A\}$, whereas if $a \in K$, then $\Gamma_{f(a)}$ contains $\neg A$ together with a sentence logically equivalent to A and is therefore inconsistent.

(b) *Theor* is a Π_2^0 set since

$$e \in \textit{Theor} \iff (\forall k)(\forall s)[(\bigwedge\{\gamma_i : i \in W_{e,s}\} \rightarrow \gamma_k) \rightarrow k \in W_e].$$

For the completeness, we give a reduction of *Inf* to *Theor*, using the restricted language with a single propositional variable A . Then we define a recursive function f such that

$$\Gamma_{f(e)} = \{\gamma_i : (\exists n)[n \in W_e \text{ \& } i < n \text{ \& } A \vdash \gamma_i]\}.$$

If $e \in \textit{Inf}$, then $\Gamma_{f(e)} = \textit{Con}(\{A\})$ and is a theory. If $e \notin \textit{Inf}$, choose n so that $m \in W_e \rightarrow m \leq n$. Then $\Gamma_{f(e)} \subseteq \{\gamma_0, \dots, \gamma_n\}$ and is thus finite. It follows that $\Gamma_{f(e)}$ is not closed under deduction, since, for example, there are infinitely many iterated conjunctions of A which are all implied by A . This proof uses only the fact that there are infinitely many consequences of any given sentence and so applies to any nonempty language.

(c) *Theor*₁ is a Π_2^0 set, since $\textit{Theor}_1 = \textit{Theor} \cap \textit{Cons}$. For the completeness, we note that the reduction f given above for *Theor* always produces a consistent set when $e \in \textit{Inf}$.

(d) *Cmpl* is a Π_2^0 set, since

$$e \in \textit{Cmpl} \iff e \in \textit{Theor}_1 \text{ \& } (\forall i, j)[\gamma_j = \neg \gamma_i \rightarrow (i \in W_e \vee j \in W_e)].$$

For the completeness, we note that the reduction given above for *Theor* produces a complete consistent theory when $e \in \textit{Inf}$. For an arbitrary language, we modify the the argument above as follows. Let Γ be a fixed complete consistent recursive theory. (This can be obtained as the theory of a trivial recursive model, with universe ω , all constants interpreted as zero, all functions interpreted as the constant zero function, and all relations interpreted as true.) Then we define a recursive function f such that

$$\Gamma_{f(e)} = \{\gamma_i \in \Gamma : (\exists n)[n \in W_e \text{ \& } i < n]\}.$$

(e) $Theor_2$ is a Σ_3^0 set, since

$$e \in Theor_2 \iff (\exists a)[a \in Rec \ \& \ a \in Cmpl \ \& \ W_e \subseteq W_a].$$

For the completeness, we define a reduction of $I_P(g\text{-bounded, rec. nonempty})$ to $Theor_2$ where $g(x) = 2$ for all x and apply Theorem 2.36. Observe that either of the proofs given above in Theorem 3.3 are uniform and let $\Gamma_{f(e)}$ be the logical theory such that the class of complete consistent extensions of $\Gamma_{f(e)}$ represents the Π_1^0 class of sets P_e . (If T_e is not a subset of $\{0, 1\}^{<\omega}$, then we ensure that $f(e) \notin Theor_2$ by putting $\gamma_i \in \Gamma_{f(e)}$ for any i such that some $\sigma \in T_e$ with $|\sigma| < i$ and with some $\sigma(t) > 1$.) Then P_e has a recursive member if and only if $\Gamma_{f(e)}$ has a decidable, complete, consistent extension. Thus

$$e \in I_P(g\text{-bounded, rec. nonempty}) \iff f(e) \in Theor_2.$$

(For an arbitrary language, simply make all the functions and all the other relations trivial.) $\square$

Next we consider index sets of decidable theories where we index a theory by the index of the total function which is its characteristic function. Give a total recursive function φ_e from ω to $\{0, 1\}$, let $\Delta_e = \{\gamma : \varphi_e(\gamma) = 1\}$.

Theorem 3.8

- (a) $Cons^* = \{e : \varphi_e \in \{0, 1\}^\omega \ \& \ \Delta_e \text{ is consistent}\}$
 $= \{e : \Delta_e \text{ has a complete consistent extension}\}$
 $= \{e : \Delta_e \text{ has a decidable, complete consistent extension}\}$

is a Π_2^0 complete set.

- (b) $Theor^* = \{e : \varphi_e \in \{0, 1\}^\omega \ \& \ \Delta_e \text{ is a theory}\}$ is a Π_2^0 complete set.

- (c) $Theor_1 = \{e \in Theor^* : \Gamma_e \text{ is a consistent theory}\}$ is a Π_2^0 complete set.

- (d) $Cmpl^* = \{e : \Gamma_e \text{ is a complete consistent theory}\}$ is a Π_2^0 complete set.

- (e) $Th_2 = \{e \in Theor_1 : \Gamma_e \text{ has a decidable complete consistent extension}\}$
is a Σ_3^0 complete set.

Proof.

(a) $Cons^*$ is a Π_2^0 set, since $e \in Cons^*$ iff

$$e \in Tot \ \& \ (\forall i) \varphi_e(i) < 2 \\ \& \ (\forall s) \neg [\wedge \{\varphi_i : i < a \ \& \ \varphi_i \in \Delta_e\} \rightarrow (\gamma_0 \wedge \neg \gamma_0)].$$

For the completeness, we define a reduction of Tot to $Cons^*$ by restricting to a language with a single fixed propositional variable A . First there is a total recursive function g such that given any partial recursive function φ_e , $\varphi_{g(e)}$ is a partial recursive function such that for all x , $\varphi_{g(e)}(x) \downarrow \iff \varphi_e(x) \downarrow$ and if $\varphi_e(x) \downarrow$, then $\varphi_{g(e)}(x) = \min(1, \varphi_e(x))$. Then define $\varphi_{f(a)}(e) = 1$ if and only if $\varphi_{g(a)}(e) \downarrow$ and $A \vdash \gamma_e$. Thus if $a \notin Tot$, then $\varphi_{f(a)} \notin Tot$, so that $e \notin Cons^*$. If $a \in Tot$, then $\varphi_{f(a)} \in Tot$ and $\Delta_{f(a)} = Con(\{A\})$ and hence is consistent.

(b) $Theor^*$ is a Π_2^0 set since $e \in Theor^*$ iff

$$e \in Tot \ \& \ (\forall k)(\forall s) [(\wedge \{\gamma_i : i < s, i \in \Delta_e\} \rightarrow \gamma_k) \rightarrow k \in \Delta_e].$$

For the completeness, we give a reduction of Tot to $Theor^*$. Let Δ be a fixed complete, consistent decidable theory and let g be a $\{0, 1\}$ -valued recursive function so that $\gamma_i \in \Delta$ if and only if $g(i) = 1$. Then we define a recursive function f such that $\varphi_{f(e)}(i) = g(i)$, if $\varphi_e(i)$ is defined. Thus $\Delta_{f(e)} = \Delta$ if $e \in Tot$ and $\varphi_{f(e)}$ is not total, otherwise.

(c) $Theor_1^*$ is a Π_2^0 set, since $Theor_1^* = Theor^* \cap Cons^*$. For the completeness, we note that the reduction f given above for $Theor$ always produces a consistent set when $e \in Tot$.

(d) $Cmpl^*$ is a Π_2^0 set, since

$$e \in Cmpl^* \iff e \in Theor_1^* \ \& \ (\forall i, j) [\gamma_j = \neg \gamma_i \rightarrow (\gamma_i \in \Delta_e \vee \gamma_j \in \Delta_e)].$$

For the completeness, we note that the reduction given above for $Theor$ produces a complete consistent theory when $e \in Inf$. $\square$

Moreover, the reductions given in Theorems 3.1 and 3.3 allows us to translate index set results for binary trees to index set results for r.e. theories. For example, parts (iii)–(vii) of Theorem 2.50 become the following results.

Theorem 3.9

(a) For any positive integer c ,

$$\left(\{e : \Gamma_e \text{ has } > c \text{ complete consistent extensions}\}, \right. \\ \left. \{e : \Gamma_e \text{ has } \leq c \text{ complete consistent extensions}\} \right)$$

is (Σ_2^0, Π_2^0) complete,

$\{e : \Gamma_e \text{ has } = c + 1 \text{ complete consistent extensions}\}$ is D_2^0 complete,
and

$\{e : \Gamma_e \text{ has } = 1 \text{ complete consistent extensions}\}$ is Π_2^0 complete.

(b) $(\{e : \Gamma_e \text{ has } \geq \aleph_0 \text{ complete consistent extensions}\},$

$$\{e : \Gamma_e \text{ has } < \aleph_0 \text{ complete consistent extensions}\})$$

is (Π_3^0, Σ_3^0) complete.

(c) $(\{e : \Gamma_e \text{ has a complete decidable consistent extension}\},$

$$\{e : \Gamma_e \text{ has no complete decidable consistent extensions}\})$$

is (Σ_3^0, Π_3^0) complete.

(d) For any positive integer c ,

$$\left(\{e : \Gamma_e \text{ has } > c \text{ complete decidable consistent extensions}\}, \right. \\ \left. \{e : \Gamma_e \text{ has } \leq c \text{ complete decidable consistent extensions}\} \right)$$

is (Σ_3^0, Π_3^0) complete, and

$\{e : \Gamma_e \text{ has } = c \text{ complete decidable consistent extensions}\}$ is D_3^0 complete,

(e) $(\{e : \Gamma_e \text{ has } < \aleph_0 \text{ complete decidable consistent extensions}\},$

$$TH_2^\star = \{e : \Gamma_e \text{ has } = \aleph_0 \text{ complete decidable consistent extensions}\})$$

is (Σ_4^0, Π_4^0) complete.

This sort of transfer of index sets results holds for many of the examples that follow. For the most part, we will not state these type of results explicitly.

4 Nonmonotonic logic

In this section, we shall show how Π_1^0 classes arise naturally in the setting of nonmonotonic logics. In fact, nonmonotonic logic is one of the few areas where all three types of Π_1^0 classes, arbitrary, bounded, and recursively bounded, arise in a natural manner. Nonmonotonic logics arose in attempts to formalize several forms of common sense reasoning. These systems include the default logic of Reiter [126], the nonmonotonic modal logics of McDermott and Doyle [99, 98], the stable semantics of general logic programs [59], and the answer sets semantics for logic programs with classical negation [60].

Classical logic, which we considered in the previous section, is monotonic in that a deduction from a set of premises remains valid for any larger set of premises. Minsky [113] suggested that there is another sort of reasoning which is not monotonic. This is the reasoning in which we deduce a statement based on the absence of any evidence against the statement. Such statements are in the category of beliefs rather than in the category of truths. Common sense or even scientific reasoning forces one to make assumptions without complete information. New information may naturally lead to the rejection of previous beliefs.

Tarski [146] characterized monotonic formal systems by means of monotonic rules of inference. Such systems include intuitionistic logic, classical logics, modal logics, and many others. Suppose that a nonempty set U is given. In a particular application U may be the collection of all formulas of a propositional or first order logic, of all legal strings of a formal system, or of all atomic statements as in logic programming.

A *monotonic rule of inference* is a tuple $r = (P, \varphi)$ where $P = (\alpha_1, \dots, \alpha_n)$ is a finite (possibly empty) list of objects from U and φ is an element of U . Such a rule r is usually written in the suggestive form

$$r = \frac{\alpha_1, \dots, \alpha_n}{\varphi}.$$

We call $\alpha_1, \dots, \alpha_n$ the premises of r , and φ the conclusion of r .

Definition 4.1

(a) A monotonic formal system is a pair (U, M) where U is a nonempty set and M is a collection of monotonic rules.

(b) A subset $S \subseteq U$ is called *deductively closed* over (U, M) if for all rules

$$r = \frac{\alpha_1, \dots, \alpha_n}{\varphi} \in M, \quad \alpha_1, \dots, \alpha_n \in S \text{ implies } \varphi \in S.$$

Inspired by Reiter [126], and Apt [5], Marek, Nerode, and Remmel developed a theory of nonmonotonic rule systems [102, 103, 104, 105, 106, 107]. Nonmonotonic rule systems are simple algebraic structures which formalize the notion of nonmonotonic reasoning. Moreover there are simple translations between nonmonotonic rule systems and each of the nonmonotonic formalisms listed above which show that theorems established for nonmonotonic rule systems immediately transfer to corresponding results for each of these nonmonotonic logics.

A *nonmonotonic rule system* is a pair (U, N) where a nonempty set U and a set N of nonmonotonic rules. A nonmonotonic rule of inference is a triple P, R, c , where $P = \{a_1, \dots, a_n\}$, $R = \{b_1, \dots, b_m\}$ are finite lists of objects from U and $c \in U$. Each such rule is written in form

$$r = \frac{a_1, \dots, a_n : b_1, \dots, b_m}{c}.$$

Here $a_1, \dots, a_n$ are called the *premises* of rule r , and $b_1, \dots, b_m$ are called the *restraints* of rule r . Either P , or R , or both may be empty. If R is empty, then the rule r is *monotonic*. If $P = G = \emptyset$, then the rule r is called an axiom. The set $\{a_1, \dots, a_n\}$ is denoted by $p(r)$, the set $\{b_1, \dots, b_m\}$ is denoted by $R(r)$, and c is denoted by $c(r)$. The intuitive idea of the nonmonotonic rule r is that c is supposed to hold if we have established that $a_1, \dots, a_n$ are true and there is no evidence that any of $b_1, \dots, b_n$ have been established.

A subset $S \subseteq U$ is called *deductively closed* if for every rule r of N , whenever all premises $a_1, \dots, a_n$ of r are in S and no restraint $b_1, \dots, b_m$ of r is in S , then the conclusion c of r belongs to S .

In a monotonic rule system, the family of deductively closed sets is closed under arbitrary intersections, so that for every $I \subseteq U$ there is the least set $T(I)$ such that $I \subseteq T(I)$ and $T(I)$ is deductively closed. The operator T is monotone, meaning that $I \subseteq J$ implies that $T(I) \subseteq T(J)$. For first order logic, $T(I) = \text{Con}(I)$. In nonmonotonic systems, the deductively closed sets are not generally closed under arbitrary intersections. But the deductively closed sets are closed under intersections of *descending* chains. Since U is deductively closed, by the Kuratowski-Zorn Lemma, any $I \subseteq U$ is contained in at least one *minimal* deductively closed set.

Given a set S and an $I \subseteq U$, an *S -deduction* of c from I in (U, N) is a finite sequence $\langle c_1, \dots, c_k \rangle$ such that $c_k = c$ and, for all $i \leq k$, each c_i is in I , or is an axiom, or is the conclusion of a rule $r \in N$ such that all the premises

of r are included in $\{c_1, \dots, c_{i-1}\}$ and all restraints of r are in $U \setminus S$. An S -consequence of I is an element of U occurring in some S -deduction from I . $C_S(I)$ is defined to be the set of all S -consequences of I in (U, N) .

Note that a monotonic rule system can be viewed as a special case of a nonmonotonic rule systems where all the rules are monotonic. In a monotonic system, $C_S(I) = T(I)$ for any subset S of U .

For a fixed S , the operator $C_S(\cdot)$ is monotonic. That is, if $I \subseteq J$, then $C_S(I) \subseteq C_S(J)$. Also, $C_S(C_S(I)) = C_S(I)$. However, $C_S(\cdot)$ is antimonotonic in S , that is, for fixed I , $S' \subseteq S$ implies that $C_S(I) \subseteq C_{S'}(I)$.

Generally, $C_S(I)$ is not deductively closed in (U, N) . It is perfectly possible to have all the premises of a rule be in $C_S(I)$, all the restraints of that rule be outside $C_S(I)$, but a restraint of that rule be in S , preventing the conclusion from being put into $C_S(I)$.

A set S is said to be an *extension* of I in (U, N) if $C_S(I) = S$. Thus in a monotonic rule system, the only extension of I is $T(I)$.

We list below some basic properties of extensions.

Proposition 4.2

- (a) *If S is an extension of I , then:*
 - (i) *S is a minimal deductively closed superset of I .*
 - (ii) *For every I' such that $I \subseteq I' \subseteq S$, $C_S(I') = S$.*
- (b) *The set of extensions of I forms an antichain. That is, if S_1, S_2 are extensions of I and $S_1 \subseteq S_2$, then $S_1 = S_2$.*

With each non-monotonic rule r , we associate a monotonic rule obtained from r by dropping all the restraints. The rule r' is called the *projection* of rule r . We write $M(S)$ for the collection of all projections of all rules from $N(S)$. The projection $(U, N) \downarrow_S$ is the monotonic rule system $(U, M(S))$. Thus $(U, N) \downarrow_S$ is obtained as follows: First, non- S -applicable rules are eliminated. Then, the restraints are dropped altogether. We have the following characterization theorem.

Theorem 4.3 *A subset $S \subseteq U$ is an extension of I in (U, N) if and only if S is the deductive closure of I in $(U, N) \downarrow_S$.*

Based on Theorem 4.3, we can give an intuitive explanation of the notion of extension for a nonmonotonic rule system. That is, one way to view an extension is that it represents a justifiable internally consistent set of beliefs given the rules of the system. The idea is to view the nonmonotonic rules of the systems as rules of thumb. One then asserts a certain set of beliefs B . Given B we eliminate all the rules which are not consistent with B , i.e., all those rules r such that $R(r) \cap B \neq \emptyset$. Then B is a justifiable internally consistent set of beliefs if we can derive everything in B from the rules that are left. On a more practical level, Theorem 4.3 tells us how to test if a collection $S \subseteq U$ is an extension of I in (U, N) .

A simple construction allows us to consider only extensions of the empty set. In fact, if $\mathcal{S}$ is a nonmonotonic rule system, and $I \subseteq U$, then the system $\mathcal{S}(I)$ arises from $\mathcal{S}$ and I by adding to N all the rules of the form $\frac{\vdots}{t}$ for all $t \in I$. We then have:

Proposition 4.4 *T is an extension of I in $\mathcal{S}$ if and only if T is an extension of $\emptyset$ in $\mathcal{S}(I)$.*

We next introduce briefly some of the nonmonotonic logical systems mentioned above and show how each can be coded into nonmonotonic rule systems.

4.1 Default Logic

Default Logic is a system based on the language $\mathcal{L}$ of propositional logic. A default rule has the form

$$r = \frac{\alpha : M\beta_1, \dots, M\beta_k}{\gamma},$$

where $\alpha, \beta_1, \dots, \beta_k, \gamma$ are formulas of $\mathcal{L}$. Following Reiter [126], a default theory is defined as a pair (D, W) where D is a set of default rules and W is a set of formulas of $\mathcal{L}$.

A theory S is said to be an *extension* of (D, W) if for all rules $r \in D$ as above, if $\alpha \in S$ and $\neg\beta_i \notin S$ for all i , then $\gamma \in S$.

(D, W) may be interpreted as a nonmonotonic rule system as follows. For every default rule r as above, construct the following nonmonotonic rule d_r .

$$d_r = \frac{\alpha : \neg\beta_1, \dots, \neg\beta_k}{\gamma}$$

Next, for every formula $\psi \in \mathcal{L}$, define the rule

$$d_\psi = \frac{\vdots}{\psi}$$

and for all pairs of formulas χ_1, χ_2 define

$$mp_{\chi_1, \chi_2} = \frac{\chi_1, \chi_1 \supset \chi_2 :}{\chi_2}.$$

Now define the set of rules $N_{D,W}$ as follows:

$$N_{D,W} = \{d_r : r \in D\} \\ \cup \{d_\psi : \psi \in W \text{ or } \psi \text{ is a tautology}\} \cup \{mp_{\chi_1, \chi_2} : \chi_1, \chi_2 \in \mathcal{L}\}.$$

It was shown in [102] that a set of formulas S is a default extension of $\langle D, W \rangle$ if and only if S is an extension of the nonmonotonic rule system $\langle U, N_{D,W} \rangle$.

4.2 Nonmonotonic modal logics

McDermott [98] introduced a technique which allows one to create nonmonotonic counterparts of various modal logics. The modal language $\mathcal{L}_L$ has one modal operator L , interpreted as necessitation, knowledge, or belief. Given a modal logic $\mathcal{S}$, McDermott defined a consequence operator $Cn_{\mathcal{S}}$ which allows for application of necessitation to all previously proven formulas, not only to the the axioms of $\mathcal{S}$.

Now, given a set of formulas $T \subseteq \mathcal{L}$ and another set of formulas I , interpreted to be the initial assumptions of the reasoning agent, a theory T is called an $\mathcal{S}$ -*expansion* of I if

$$T = Cn_{\mathcal{S}}(I \cup \{\neg L\psi : \psi \notin T\}).$$

The set of formulas $\{\neg L\psi : \psi \notin T\}$ represents the so-called “negative introspection with respect to T ”. The modal logic $\mathcal{S}$ may be simulated as a nonmonotonic rule system as follows. The universe U will be as before the set of all well formed formulas of the language $\mathcal{L}_L$.

For every formula $\psi \in \mathcal{L}_L$ we consider a rule:

$$e_\psi = \frac{\vdots \psi}{\neg L\psi}.$$

Now, given a theory I (the set of initial assumptions) in a modal logic $\mathcal{S}$, and a theory $T \subseteq \mathcal{L}_L$ consider the following set of rules

$$N_{I,\mathcal{S}} = \{d_\psi : \psi \in \mathcal{S}\} \cup \{e_\psi : \psi \in \mathcal{L}_L\} \cup \{mp_{\chi_1, \chi_2} : \chi_1, \chi_2 \in \mathcal{L}_L\} \\ \cup \{d_\psi : \psi \in I\} \cup \{d_\psi : \psi \text{ is a tautology of } \mathcal{L}_L\}$$

It may then be seen that T is an $\mathcal{S}$ -expansion of I if and only if T is an extension of the nonmonotonic rule system $(U, N_{I,\mathcal{S}})$.

4.3 General logic programming

General logic programs extend the usual syntax of (Horn) logic programs by admitting negated atoms in the body of clauses. Specifically, a general clause is an expression of the form:

$$C = p \leftarrow q_1, \dots, q_m, \neg r_1, \dots, \neg r_n.$$

Here we only assume that $m \geq 0$ and $n \geq 0$ so that the usual logic programming clauses are special cases of general clauses. General clauses possess the *logical interpretation*:

$$q_1 \wedge \dots \wedge q_m \wedge \neg r_1 \wedge \dots \wedge \neg r_n \supset p.$$

As long as we are interested in Herbrand models of general programs, we can consider the propositional theory $ground(P)$ consisting of all ground substitutions of clauses of P . While P is usually finite, $ground(P)$ may be infinite if P contains function symbols. There is of course a one-to-one correspondence between Herbrand models of P and propositional models of $ground(P)$.

As is the case for (Horn) logic programming, not every model of the program has a clear computational meaning. Some models of a general program provide a computationally sound meaning to negation. We shall discuss here only the stable models of Gelfond and Lifschitz [59] since stable models are most naturally modeled by extensions of nonmonotonic rule systems.

Given a subset M of the Herbrand universe, and a clause C as above in $ground(P)$, we define C^M as *nil* if $r_j \in M$ for some $1 \leq j \leq n$. Otherwise $C^M = p \leftarrow q_1, \dots, q_m$. Then we put

$$P^M = \{C^M : C \in ground(P)\}.$$

Since P^M is a Horn program, it possesses a least Herbrand model, N_M . Then we call M a *stable model* of P if $M = N_M$. It is easy to see that a stable model of P is a model of P . The stable models of logic programs may be encoded as extensions of nonmonotonic rule systems as follows. The universe of all our system, U , will be the Herbrand base of the program. Next, to every general propositional clause C as above, assign the rule

$$r_C = \frac{q_1, \dots, q_m : r_1, \dots, r_n}{p}.$$

Now, given the program P , define

$$N_P = \{r_C : C \in \text{ground}(P)\}.$$

Then M is a stable model of P if and only if M is an extension of the nonmonotonic rule system (U, N_P) .

4.4 Proof Schemes

We now return to studying the complexity of the set of extensions of nonmonotonic rule systems. The basic notion used to analyze the Turing complexity of the set of extensions of nonmonotonic rule systems is that of a proof scheme.

A *proof scheme* s is a finite sequence of triples

$$((c_1, r_1, Z_1), \dots, (c_n, r_n, Z_n))$$

where $c_1, \dots, c_n \in U$, $r_1, \dots, r_n \in N$, $Z_1, \dots, Z_n$ are finite subsets of U such that for all $1 \leq j \leq n$,

- (1) $c_1 = c(r_1)$, $Z_1 = R(r_1)$ and $p(r_1) = \emptyset$
- (2) For $j > 1$, $p(r_j) \subseteq \{c_1, \dots, c_{j-1}\}$, $c(r_j) = c_j$, and $Z_j = Z_{j-1} \cup R(r_j)$.
- (3) c_n is the conclusion of s and is denoted by $cln(s)$. Z_n is called the support of s and is denoted by $supp(s)$.

Clearly an initial segment of a proof scheme is also a proof scheme.

Notice that the support of a proof scheme s , Z_n , has the property that for every set S such that $S \cap Z_n = \emptyset$, the sequence $(c_1, \dots, c_n)$ is an S -derivation. Conversely if $(c_1, \dots, c_n)$ is an S -derivation, then there is a proof scheme

$$s = ((c_1, r_1, Z_1), \dots, (c_n, r_n, Z_n))$$

such that $Z_n \cap S = \emptyset$.

There is a natural preordering on proof schemes. Namely, $s_1 \prec s_2$ if and only if every rule appearing in s_1 appears in s_2 as well. The relation $\prec$ is not a partial ordering but it is well-founded. We can thus talk about minimal proof schemes for a given element $c \in U$. Intuitively, a minimal proof scheme carries the minimal information necessary to derive its conclusion. Since the support of every proof scheme is finite, the negative information carried in such a proof scheme is finite.

Proof schemes can be used to characterize extensions. We say that a set S admits a proof scheme s if $\text{supp}(s) \cap S = \emptyset$. We then have the following characterization of extensions.

Proposition 4.5 *Let $\mathcal{S} = (U, N)$ be a nonmonotonic rule system. Let $S \subseteq U$. Then S is an extension of $\mathcal{S}$ if and only if the following conditions are met:*

- (a) *If s is a proof scheme and S admits s , then $c(s) \in S$.*
- (b) *Whenever $a \in S$ then there exists a proof scheme s such that S admits s .*

It is easy to see that we can restrict to minimal proof schemes in Proposition 4.5.

4.5 Π_1^0 Classes and extensions

We now give the basic results from [102, 103, 104, 106] on the complexity of extensions in recursive nonmonotonic rule systems.

The canonical index, $\text{can}(X)$, of the finite set $X = \{x_1 < \dots < x_n\} \subseteq \omega$ is defined as $2^{x_1} + \dots + 2^{x_n}$ and the canonical index of $\emptyset$ is defined as 0. Let D_k be the finite set whose canonical index is k , i.e., $\text{can}(D_k) = k$.

Let (U, N) be a nonmonotonic rule system where $U \subseteq \omega$. We shall identify a rule r with the code of a triple $\langle k, l, \varphi \rangle$ where $D_k = p(r)$, and $D_l = R(r)$, $\varphi = c(r)$. In this way we can think about N as a subset of ω as well. This given, we then say that an NRS $\mathcal{S} = (U, N)$ is *recursive* if U and N are recursive subsets of ω .

There are two important subclasses of recursive NRS's introduced in [104], namely locally finite and highly recursive nonmonotonic rules systems. We

say that the system (U, N) is *locally finite* if for each $c \in U$, there are only finitely many $\prec$ -minimal proof schemes with conclusion c . Given a proof scheme for c , $s = (c_1, r_1, Z_1), \dots, (c_n, r_n, Z_n)$, the code of s , $c(s)$, is defined by

$$c(s) = \langle \langle c_1, r_1, Z_1 \rangle, \dots, \langle c_n, r_n, Z_n \rangle \rangle.$$

If $\langle U, N \rangle$ is a locally finite recursive nonmonotonic rule system and $c \in U$, we let Dr_c denote the set of codes of all $\prec$ -minimal proof schemes for c . We say that (U, N) is *highly recursive* if (U, N) is recursive, locally finite, and the map $c \mapsto \text{can}(Dr_c)$ is partial recursive. The latter means that there is an effective procedure which, when applied to any $c \in U$, produces a canonical index of the set of all codes of $\prec$ -minimal proof schemes with conclusion c .

The following results are due to Marek, Nerode, and Remmel [104].

Theorem 4.6 *For any highly recursive NRS system $\mathcal{S} = (U, N)$, there is a highly recursive tree $T_{\mathcal{S}}$ such that there is an effective one-to-one degree preserving correspondence between $[T_{\mathcal{S}}]$ and $\mathcal{E}(\mathcal{S})$.*

Vice versa, for any highly recursive tree T , there is a highly recursive NRS system $\mathcal{S}_T = (U, N)$ such that there is an effective one-to-one degree preserving correspondence between $[T]$ and $\mathcal{E}(\mathcal{S}_T)$.

Theorem 4.7 *For any locally finite recursive NRS system $\mathcal{S} = (U, N)$, there is a finitely branching recursive tree $T_{\mathcal{S}}$ such that there is an effective one-to-one degree preserving correspondence between $[T_{\mathcal{S}}]$ and $\mathcal{E}(\mathcal{S})$.*

Vice versa, for any highly recursive tree T in $\mathbf{0}'$, there is a locally finite recursive NRS system $\mathcal{S}_T = (U, N)$ such that there is an effective one-to-one degree preserving correspondence between $[T]$ and $\mathcal{E}(\mathcal{S}_T)$.

Theorem 4.8 *For any recursive NRS system $\mathcal{S} = (U, N)$, there is a recursive tree $T_{\mathcal{S}}$ such that there is an effective one-to-one degree preserving correspondence between $[T_{\mathcal{S}}]$ and $\mathcal{E}(\mathcal{S})$.*

Vice versa, for any recursive tree T , there is a recursive NRS system $\mathcal{S}_T = (U, N)$ such that there is an effective one-to-one degree preserving correspondence between $[T]$ and $\mathcal{E}(\mathcal{S}_T)$.

As usual, we can immediately derive a number of corollaries about the complexity of the set of extensions of a recursive nonmonotonic rule systems by transferring known results about Π_1^0 -classes. For example, for recursive nonmonotonic rule systems, we have the following results, see [104].

Corollary 4.9

- (a) Every recursive NRS system $\mathcal{S} = (U, N)$ which has an extension has an extension E such that $E \leq_T B$ where B is a complete Π_1^1 -set.
- (b) If $\mathcal{S} = (U, N)$ is a recursive NRS system with a unique extension E , then E is hyperarithmetic.

Corollary 4.10

- (a) There is a recursive NRS system $\mathcal{S} = (U, N)$ such that $\mathcal{S}$ has an extension but $\mathcal{S}$ has no extension which is hyperarithmetic.
- (b) For each recursive ordinal α , there exists a recursive NRS system $\mathcal{S} = (U, N)$ possessing a unique extension E such that $E \equiv_T \mathbf{0}^{(\alpha)}$.

Corollary 4.11 Let $\mathcal{S} = (U, N)$ be a highly recursive nonmonotonic rule system such that $\mathcal{E}(\mathcal{S}) \neq \emptyset$. Then,

- (a) There exists an extension E of $\mathcal{S}$ such that E is low.
- (b) If $\mathcal{S}$ has only finitely many extensions, then every extension E of $\mathcal{S}$ is recursive.

In the other directions, there are a number of corollaries of Theorem 4.6 which allow us to show that there are highly recursive NRS systems $\mathcal{S}$ such that the set of degrees realized by elements of $\mathcal{E}(\mathcal{S})$ are still quite complex. Again all these corollaries follow by transferring results of Section 2.

Corollary 4.12

- (a) There is a highly recursive nonmonotonic rule system (U, N) which has $2^{\aleph_0}$ extensions but has no recursive extensions.
- (b) There is a highly recursive nonmonotonic rule system (U, N) such that (U, N) has $2^{\aleph_0}$ extensions and any two extensions $E_1 \neq E_2$ of (U, N) are Turing incomparable.
- (c) There is a highly recursive nonmonotonic rule system (U, N) such that (U, N) has $2^{\aleph_0}$ extensions and if $\mathbf{a}$ is the degree of any extension E of (U, N) and $\mathbf{b}$ is any recursively enumerable degree such that $\mathbf{a} <_T \mathbf{b}$, then $\mathbf{b} \equiv_T \mathbf{0}'$.

- (d) If $\mathbf{a}$ is any recursively enumerable Turing degree, then there is a highly recursive nonmonotonic rule system (U, N) such that (U, N) has $2^{\aleph_0}$ extensions and the set of recursively enumerable degrees $\mathbf{b}$ which contain an extension of (U, N) is precisely the set of all recursively enumerable degrees $\mathbf{b} \geq_T \mathbf{a}$.

Finally, we note that there are analogues of Corollaries 4.11 and 4.12 which hold for recursive locally finite nonmonotonic rule systems. That is, one can replace highly recursive nonmonotonic rule systems by recursive locally finite nonmonotonic rule systems if one replaces all the statements about degrees of extensions by the corresponding statement relative to an $\mathbf{0}'$ oracle. For example, the analogue of part (1) of Corollary 4.11 is that every recursive locally finite nonmonotonic rule system $\mathcal{S}$ such that $\mathcal{E}(\mathcal{S}) \neq \emptyset$ has an extension E such that the jump of E is recursive in $\mathbf{0}''$, while the analogue of (1) of Corollary 4.12 is that there exists a recursive locally finite nonmonotonic rule system (U, N) which has $2^{\aleph_0}$ extensions but which has no extension which is recursive in $\mathbf{0}'$. Moreover, we can weaken the hypothesis of locally finite and highly recursive slightly and still derive the same theorems. That is, we say that a recursive nonmonotonic rule system (U, N) has the *finite support property* if for each $c \in U$, the set of supports of all $\prec$ -minimal proof schemes of c is finite. It is possible for a $c \in U$ to have infinitely many $\prec$ -minimal proof schemes with the same support so that not every recursive nonmonotonic rule system with the finite support property is locally finite. Similarly, we say that a recursive nonmonotonic rule system (U, N) which has the finite support property has the *recursive finite support property* if there is an effective algorithm which given any $c \in U$, produces the canonical index of the set of canonical indices of the supports of all the $\prec$ -minimal proof schemes of c . See [106] for further details. Finally there are complete analogues of all the results of this section which apply to finite predicate logic programs or finite predicate logic default theories, see Section 4.7.

4.6 Extended Nonmonotonic Rule Systems

An *extended* nonmonotonic rule system permits rules of the form

$$r = \frac{a_1, \dots, a_m : Z}{c},$$

where $a_1, \dots, a_m \in U$, $Z \subseteq U$, $c \in U$.

Ferry [50, 51] proved that many properties of nonmonotonic rule systems generalize to such extended nonmonotonic rule systems.

If Z is a recursive set, then it may be coded by e such that $\varphi_e = \chi_Z$. Then the rule above may be coded by a quadruple $c(r) = [k, e, h, c]$ such that k codes the finite set $\{a_1, \dots, a_n\}$ and $h = 2 + \max(Z)$ if Z is finite, $h = 0$ if Z is infinite.

If $\mathcal{S} = (U, N)$ is an extended monotonic rule system with $U \subseteq \omega$, then the pair $(\mathcal{S}, \mathcal{C}(N))$ is said to be a *recursive extended nonmonotonic rule system* if U is a recursive subset of ω and $\mathcal{C}(N)$ is a recursive set of codes for rules in N containing precisely one code for each rule r .

Proof schemes for recursive extended nonmonotonic systems are defined as before and may be coded using the codes for rules given above. $(\mathcal{S}, \mathcal{C}(N))$ is *locally finite* if for every $a \in U$ there are finitely many $\prec$ -minimal proof schemes in $\mathcal{S}$ with conclusion a . We let Dr_a denote the set of all codes of $\prec$ -minimal proof schemes for a in $\mathcal{S}$. Then we say that $(\mathcal{S}, \mathcal{C}(N))$ is a *highly recursive* ENRS if $(\mathcal{S}, \mathcal{C}(N))$ is a locally finite ENRS and the map $a \mapsto \text{can}(Dr_a)$ is partial recursive.

The expressive power of recursive extended nonmonotonic rule systems is shown by the following theorems from [107].

Theorem 4.13 *Let T be a recursive tree, $T \subseteq \omega^{<\omega}$. Then there is a highly recursive ENRS $(\mathcal{S}, \mathcal{C}(N))$ where $\mathcal{S} = (U, N)$ such that there exists an effective, one-to-one, degree-preserving correspondence between $\mathcal{E}(\mathcal{S})$ and the set $[T]$.*

Theorem 4.14 *Let $(\mathcal{S}, \mathcal{C}(N))$ be a recursive extended nonmonotonic rule system where $\mathcal{S} = (U, N)$. Then there is a recursive tree $T \subseteq \omega^{<\omega}$ and a pair of indices e and f such that*

- (a) *for any $\pi \in [T]$, $\varphi_e^\pi = \chi_{E_\pi}$ (where E_π is an extension of $\mathcal{S}$ and $E_\pi \leq_T \pi$), and*
- (b) *for any extension E of $\mathcal{S}$, $\varphi_f^{E'} = \chi_{\pi_E}$, where $\pi_E \in [T]$ and $\pi_E \leq_T E'$ (where E' is the jump of E).*

Moreover for all $E \in \mathcal{E}(\mathcal{S})$, $E_{\pi_E} = E$, and for all $\pi \in [T]$, $\pi_{E_\pi} = \pi$.

Note that all our theorem says is that $E \mapsto \pi_E$ is an effective, one-to-one correspondence between $\mathcal{E}(\mathcal{S})$ and $[T]$ but it only satisfies the inequality $E \leq_T \pi_E \leq_T E'$ rather than $\pi_E \equiv_T E$ as in the effective one-to-one degree preserving correspondence constructed in Theorem 4.13.

This still implies most of the results one would get from the stronger equivalence. For example, if (U, N) has a unique extension E , then E is hyperarithmetical.

Our final result will show that Theorem 4.14 cannot be strengthened to ensure that for any recursive ENRS $\mathcal{S} = (U, N)$, there exists a recursive tree T and an effective, one-to-one degree-preserving correspondence between $\mathcal{E}(\mathcal{S})$ and $[T]$ (as is the case of Theorem 4.8 for nonmonotonic rule systems).

Theorem 4.15 *There exists a recursive extended nonmonotonic rule system $\mathcal{S} = (U, N)$ such that there is no recursive tree $T \subseteq \omega^{<\omega}$ such that there is an effective one-to-one degree-preserving correspondence between $\mathcal{E}(\mathcal{S})$ and $[T]$.*

4.7 Predicate Logic Programs

We end this section with an extension of the results of the previous section to finite predicate logic programs. In this setting, we get a perfect correspondence between Π_1^0 classes and the set of stable models of finite predicate logic program. That is, given any finite predicate logic program P , there is a recursive tree T_P such that there is an effective one-to-one correspondence between the set of stable models of P and the paths through T_P . Vice versa, given any recursive tree T , there is a recursive program P_T such that there is an effective one-to-one correspondence between the set of stable models of P_T and the paths through T . Moreover under these correspondences, bounded trees correspond to a natural set of finite predicate logic programs called finite support property programs FSP and r.b. programs correspond to recursively FSP programs. These correspondences can be found in [106] and they essentially allow us to translate all the results on index sets for trees to results on index sets for finite predicate logic programs.

For an introductory treatment of Predicate Logic Programs, see [97]. Here is a brief self-contained account of their stable models [59]. Assume as given a fixed first order language based on predicate letters, constants, and function symbols. The Herbrand base of the language is defined as the set B_L of all ground atoms (atomic statements) of the language. A literal is an atomic formula or its negation, a ground literal is an atomic statement or its negation. A Logic Program P is a set of “program clauses”, that is, an expression of the form:

$$p \leftarrow l_1, \dots, l_k$$

where p is an atomic formula, and $l_1, \dots, l_k$ is a list of literals.

Then p is called the conclusion of the clause, the list $l_1, \dots, l_k$ is called the body of the clause. Ground clauses are clauses without variables. Horn clauses are clauses with no negated literals, that is, with atomic formulas only in the body. Horn clause programs are programs P consisting of Horn clauses. Each such program has a least model in the Herbrand base determined as the least fixed point of a continuous operator T_P representing 1-step Horn clause logic deduction ([97]).

A ground instance of a clause is a clause obtained by substituting ground terms (terms without variables) for all variables of the clause. The set of all ground instances of the program P is called $ground(P)$.

Let M be any subset of the Herbrand base. A ground clause is said to be M -applicable if the atoms whose negations are literals in the body are not members of M . Such clause is then *reduced* by eliminating remaining negative literals. This monotonicization $GL(P, M)$ of P with respect to M is the propositional Horn clause program consisting of reducts of M -applicable clauses of $ground(P)$ (see Gelfond-Lifschitz [59]). Then M is called a *stable model* for P if M is the least model of the Horn clause program $GL(M, P)$. We denote this least model as $N_{M,P}$. It is easy to see that a stable model for P is a minimal model of P [59]. We denote by $Stab(P)$ the set of all stable models of P . There may be no, one, or many stable models of P .

A *proof scheme* for p with respect to P is a sequence of triples

$$((p_l, C_l, S_l))_{1 \leq l \leq n},$$

with n a natural number, such that the following conditions all hold.

- (1) Each p_l is in $B_{\mathcal{L}}$. Each C_l is in $ground(P)$. Each S_l is a finite subset of $B_{\mathcal{L}}$.
- (2) p_n is p .
- (3) The S_l, C_l satisfy the following conditions. For all $1 \leq l \leq n$, one of (a), (b), (c) below holds.
 - (a) C_l is $p_l \leftarrow$, and S_l is S_{l-1} ,
 - (b) C_l is $p_l \leftarrow \neg s_1, \dots, \neg s_r$ and S_l is $S_{l-1} \cup \{s_1, \dots, s_r\}$, or
 - (c) C_l is $p_l \leftarrow p_{m_1}, \dots, p_{m_k}, \neg s_1, \dots, \neg s_r$, $m_1 < l, \dots, m_k < l$, and S_l is $S_{l-1} \cup \{s_1, \dots, s_r\}$.

(We put $S_0 = \emptyset$).

Suppose that $\varphi = ((p_l, C_l, S_l))_{1 \leq l \leq n}$ is a proof scheme. Then $\text{conc}(\varphi)$ denotes atom p_n and is called the *conclusion* of φ . Also, $\text{supp}(\varphi)$ is the set S_n and is called the *support* of φ .

Condition (3) tells us how to construct the S_l inductively, from the S_{l-1} and the C_l . The set S_n consists of the negative information of the proof scheme.

Formally, preorder proof schemes φ, ψ by $\varphi \prec \psi$ if

- (1) φ, ψ have same conclusion,
- (2) Every clause in φ is also a clause of ψ .

The relation $\prec$ is reflexive, transitive, and well-founded. Minimal elements of $\prec$ are minimal proof schemes.

We can characterize stable models via proof schemes as follows.

Proposition 4.16 *Let P be a program. Also, suppose that M is a subset of the Herbrand universe $B_{\mathcal{L}}$. Then M is a stable model of P if, and only if, for every $p \in B_{\mathcal{L}}$, it is true that p is in M if and only if there exists a proof scheme φ with conclusion p such that the support of φ is disjoint from M .*

A *finitary support* program (FSP program) is a Logic Program such that for every atom p , there is a finite set of finite sets S , which are exactly the inclusion-minimal supports of all those minimal proof schemes with conclusion p .

A *recursively* FSP program is an FSP recursive program such that we can uniformly compute the finite family of supports of proof schemes with conclusion p from p . The meaning of this is obvious, but we need a technical notation for the proofs.

Start by listing the whole Herbrand base of the program, $B_{\mathcal{L}}$ as a countable sequence in one of the usual effective ways. This assigns an integer (Gödel number) to each element of the base, its place in this sequence. This encodes finite subsets of the base as finite sets of natural numbers, all that is left is to code each finite set of natural numbers as a single natural number, its *canonical index*. We also set $\text{can}(\emptyset) = 0$. If program P is FSP, and the list,

in order of magnitude, of Gödel numbers of all minimal support of schemes with conclusion p is

$$Z_1^p, \dots, Z_{l_r}^p,$$

then define a function $su^P: B_{\mathcal{L}} \rightarrow \omega$ as below.

$$p \mapsto \text{can}(\{\text{can}(Z_1^p), \dots, \text{can}(Z_{l_r}^p)\})$$

We call a Logic Program P a *recursively FSP program* if it is FSP and the function su^P is recursive.

In [106], Marek, Nerode, and Remmel proved the following two results.

Theorem 4.17 *We suppose that the first order language $\mathcal{L}$ has infinitely many ground atoms.*

- (a) *Then for any recursive program P in $\mathcal{L}$, there exists a recursive tree $T \subseteq \omega^{<\omega}$ and an effective one-to-one degree preserving correspondence between the set of all stable models of P , $\text{Stab}(P)$ and $[T]$, the set of all infinite paths through T .*
- (b) *If, in addition to the hypothesis of (1), program P is FSP, then the tree T is bounded.*
- (c) *If, in addition to the hypothesis of (2), program P is recursively FSP, then the tree T is a highly recursive tree.*

Theorem 4.18 *Let C be any Π_1^0 -class. Then,*

- (a) *There is a finite program, P , and an effective one-to-one degree preserving correspondence between the elements of C and the set of all stable models of P , $\text{Stab}(P)$.*
- (b) *If in addition, C is of the form $[T]$ for a bounded recursive tree T , then P can be chosen FSP.*
- (c) *If in addition, T is a highly recursive tree, then P can be chosen recursively FSP.*

These two results were strengthened by Cenzer, Marek, and Remmel [19] to prove the following.

Theorem 4.19 *We suppose that the first order language $\mathcal{L}$ has infinitely many ground atoms.*

- (a) *Then for any finite predicate logic program P in $\mathcal{L}$, there exists a primitive recursive tree $T \subseteq \omega^{<\omega}$ and an effective one-to-one degree preserving correspondence between the set of all stable models of P , $\text{Stab}(P)$ and $[T]$, the set of all infinite paths through T .*
- (b) *If, in addition to the hypothesis of (1), the program P is FSP, then the tree T is bounded.*
- (c) *If, in addition to the hypothesis of (2), the program P is recursively FSP, then the tree T is a highly recursive tree.*

Theorem 4.20 *Let T be any primitive recursive tree. Then,*

- (a) *There is a finite program, P , and an effective one-to-one degree preserving correspondence between the elements of $[T]$ and the set of all stable models of P , $\text{Stab}(P)$.*
- (b) *If in addition, T is bounded, then P is FSP.*
- (c) *If in addition, T is a highly recursive tree, then P is recursively FSP.*

The crucial point about the proof of Theorems 4.19 and 4.20 is that they are completely uniform. For example, given a finite predicate logic program P , we can uniformly find the index of a primitive recursive tree T_P such that there is an effective one-to-one degree preserving correspondence between the stable models of P and the elements of $[T_P]$. Vice versa, given any primitive recursive tree T , we can uniformly find a finite predicate logic program P_T such that there is an effective one-to-one degree preserving correspondence between the stable models of P_T and the elements of $[T]$. This means that one can transfer all the index set results about trees and Π_0^1 classes to index sets about finite predicate logic programs. Thus if we fix some recursive first order language $\mathcal{L}$ with infinitely many ground atoms, then we can effectively list all finite predicate logic programs $P_0, P_1, \dots$. Then for any property $\mathcal{R}$, we can define an index set

$$\text{Prog}(\mathcal{R}) = \{e : P_e \text{ has property } \mathcal{R}\}.$$

We can then transfer all the index set results to index set results for finite predicate logic programs. For example, the translation of parts (3) and (5) Theorem 2.33 become the following.

Theorem 4.21

- (a) $\text{Prog}(\text{FSP}) = \{e : P_e \text{ has the FSP property}\}$ is a Σ_3^0 complete set.
- (b) $\text{Prog}(\text{recursively FSP}) = \{e : P_e \text{ has the recursively FSP property}\}$ is a Π_3^0 complete set.

Similarly, parts (2)–(4) of Theorem 2.36 become the following

Theorem 4.22

- (a) $\text{Prog}(\text{recursively FSP and has } \geq \aleph_0 \text{ stable models})$ is D_3^0 complete, and $\text{Prog}(\text{recursively FSP and has } < \aleph_0 \text{ stable models})$ is Σ_3^0 complete.
- (b) $\text{Prog}(\text{is FSP and has } \geq \aleph_0 \text{ stable models})$ is Π_4^0 complete, and $\text{Prog}(\text{is FSP and has } < \aleph_0 \text{ stable models})$ is Σ_4^0 complete.
- (c) $(\text{Prog}(\text{has } \geq \aleph_0 \text{ stable models}), \text{Prog}(\text{has } < \aleph_0 \text{ stable models}))$ is (Σ_1^1, Π_1^1) complete.

See [19] for further details.

5 Recursive algebra

In this section, we will consider the following algebraic problems.

- (1) The problem of finding a prime ideal for a commutative ring with unity.
- (1') The problem of finding a maximal ideal for a commutative ring with unity.
- (2) The problem of finding a maximal subgroup of an Abelian group.
- (3) The problem of finding a maximal ideal for a Boolean algebra.

We shall also consider the related simpler problems.

- (1'') The problem of finding a proper ideal for a commutative ring with unity.
- (2') The problem of finding a proper subgroup of an Abelian group.
- (3') The problem of finding a proper ideal for a Boolean algebra.

A recursively presented group, ring, or field consists of a recursive subset U of ω , the universe of the structure, together with appropriate partial recursive functions over U for addition, subtraction, multiplication and/or division functions as required. Unless, explicitly stated otherwise, we will assume that all our structures are countably infinite so that there is no loss in generality in assuming that the underlying universe is ω . An *r.e. ring* is the quotient of a recursive ring modulo an r.e. ideal, an *r.e. group* is the quotient of a recursive group modulo an r.e. normal subgroup, and an *r.e. Boolean Algebra* is the quotient of a recursive Boolean Algebra modulo an r.e. ideal.

We will show that the set of prime ideals of an r.e. commutative ring with unity and the set of prime ideals of an r.e. Boolean algebra can always be represented by an r.b. Π_1^0 class. We will show that the set of maximal ideals of an r.e. commutative ring with unity and the set of maximal subgroups of an r.e. group can always be represented by a Π_2^0 class. We shall also show that the set of all ideals or the set of all maximal ideals of a recursive Boolean algebra can be represented as the set of paths through a recursive tree with no dead ends.

Reversing such results, we will show that any r.b. bounded Π_1^0 class can be strongly represented by the set of maximal ideals of an r.e. Boolean algebra. We show that the set of paths through any recursive tree T with no dead ends can be represented as the set of maximal ideals of a recursive Boolean algebra. We shall also show that the set of separating sets $S(A, B)$ of a pair of r.e. sets can be represented by the set of prime ideals or the set of maximal ideals of an r.e. commutative ring with identity.

We refer the reader to Downey [38] for a general survey of recursive algebra.

We begin with a brief discussion of the various problems.

Ideals and Subgroups.

Recall that a subset H of an Abelian group $G = (G, +^G, -^G, 0^G)$ is a subgroup if it satisfies the following conditions:

$$(i) \quad 0^G \in H.$$

$$(ii) \quad a \in H \text{ and } b \in H \text{ implies } a -^G b \in H.$$

H is a *maximal subgroup* if, in addition, there is no subgroup J of G such that $H \subset J \subset G$.

A subset I of a commutative ring with unit $R = (R, +^R, -^R, \cdot^R, 0^R, 1^R)$ is an ideal I if it is a subgroup of $R = (R, +^R, 0^R)$ and it satisfies the following additional conditions:

$$(iii) \quad a \in I \text{ and } r \in R \text{ implies } a \cdot^R b \in I.$$

$$(iv) \quad 1^R \notin I.$$

I is a *prime ideal* if, in addition,

$$(v) \quad a \cdot^B b \in I \text{ implies } a \in I \text{ or } b \in I.$$

I is a *maximal ideal* if, in addition, there is no ideal J such that $I \subset J$. It is easy to see that any maximal ideal is prime, but the converse is not always true.

The classical results that every proper subgroup of a group has an extension to a maximal (and therefore proper) subgroup and that every ideal in a ring has an extension to a maximal (and therefore prime) ideal follow easily from Zorn's Lemma. In particular, if the commutative ring R with unity is not a field, then R has, for each non-unit a a proper ideal $Ra = \{ra : r \in R\}$ and therefore has a maximal ideal.

Any Boolean algebra $(B, \vee^B, \wedge^B, \neg^B, 0^B, 1^B)$ may be viewed as a commutative ring with unity where

$$a \cdot^B b = a \wedge^B b, \quad \text{and}$$

$$a + b = (a \wedge^B \neg^B b) \vee^B (\neg^B a \wedge^B b).$$

In a Boolean ring any prime ideal is maximal, so it follows from the Boolean algebra results that, for any Π_1^0 class P , there is an r.e. commutative ring with unity such that $\text{Max}(R) = \text{Prime}(R)$ is represented by P . However, there turns out to be a significant difference between Boolean rings and rings in general. The proof that any recursive Boolean ring has a recursive maximal ideal cannot be extended to arbitrary rings and in fact, a recursive ring need not have a recursive maximal ideal. This naturally led to the conjecture that any Π_1^0 class could be represented as the set of prime ideals of some commutative ring. By considering rings of polynomials, Friedman-Simpson-Smith obtained in [54] the partial result that any Π_1^0 class of separating sets can be represented as the set of prime ideals of some recursive commutative ring with unity.

Boolean algebras.

A recursive Boolean algebra $\mathcal{B} = (B, \wedge^B, \vee^B, \neg^B, \leq^B, 0^B, 1^B)$ is given by a recursive set B together with recursive binary operations $\wedge^B$ (meet) and $\vee^B$ (join) and a recursive unary operation $\neg^B$. These operations define a recursive partial ordering $\leq^B$, where $a \leq^B b$ if and only if $a \vee^B b = b$ (or equivalently $a \wedge^B b = a$.) A full development of recursive and recursively enumerable Boolean algebras may be found in [127, 130].

A subset I of B is said to be an ideal if it satisfies the following conditions:

- (i) $a \in I$ and $b \in I$ implies $a \vee^B b \in I$.
- (ii) $a \in I$ and $b \in B$ implies $a \wedge^B b \in I$.
- (iii) $1 \notin I$.

I is a *prime ideal* if

- (iv) $a \cdot^B b \in I$ implies $a \in I$ or $b \in I$.

Finally, I is a *maximal ideal* if

- (v) For any $a \in B$, $a \in I$ or $\neg^B a \in I$.

An ideal of a Boolean algebra is prime if and only if it is maximal. The classical result here is the Boolean prime ideal theorem which states that every ideal has an extension to a prime ideal.

A subset F of $\mathcal{B}$ is a *filter* if $\{a : \neg^B a \in F\}$ is an ideal. This demonstrates that the set of filters (respectively, maximal filters) of a Boolean algebra $\mathcal{B}$ is recursively isomorphic to the set of ideals (resp. maximal ideals) of $\mathcal{B}$.

There is a natural connection between first-order theories and Boolean algebras given by the *Lindenbaum algebra* $L(\Gamma)$ of a theory. The elements of $L(\Gamma)$ are the equivalence classes of $Sent(\Gamma)$ under the relation where ψ and φ are equivalent if and only if $\psi \iff \varphi \in \Gamma$ and the operations of meet, join and complement are given respectively by the conjunction, disjunction and negation. Thus if Γ is a decidable theory, $L(\Gamma)$ is a recursive Boolean algebra and if Γ is a axiomatizable theory, $L(\Gamma)$ is an r.e. Boolean algebra.

If the underlying language is propositional, then the underlying Boolean algebra $\mathcal{B}(\mathcal{L})$ may be viewed as the Lindenbaum algebra of the set of tautologies so that the equivalence is truth-table equivalence. If in addition, $\mathcal{L}$ is an effective language, it follows from the usual truth-table algorithm that $\mathcal{B}(\mathcal{L})$ is a recursive Boolean algebra. (A representative for the equivalence

class of φ may be chosen from the set of sentences of length $\leq |\varphi|$.) In this Lindenbaum algebra, it is clear that consistent theories corresponds to filters and maximal filters to complete consistent theories.

Theorem 5.1

- (a) For any r.e. commutative ring R with unity, the set of all ideals of R and the set of prime ideals of R can be represented by r.b. Π_1^0 classes, and the set of maximal ideals of R can be represented by a Π_2^0 class.
- (b) For any r.e. group G , the set of all subgroups of G can be represented by an r.b. Π_1^0 class, and the set of maximal subgroups of G can be represented by a Π_2^0 class.
- (c) For any r.e. Boolean algebra B , the set of all ideals of B and the set of prime ideals of B can be represented by r.b. Π_1^0 classes.

Proof.

(a) Let A be the underlying recursive ring and I the r.e. ideal such that R is the quotient A/I . Then there is an effective one-to-one correspondence between the ideals of R and the ideals of A which extends I , defined as follows. For any ideal J of A which extends I , let $J_R = \{[a] \in R : a \in J\}$. Similarly, given an ideal J_R of R , let $J = \{a \in A : [a] \in J_R\}$. Since $0 \in J$, it follows that $I \subset J$. It is easy to see that J_R is prime if and only if J is prime and is maximal if and only if J is maximal. Thus we will actually consider the Π_1^0 class $\text{Prime}(A, I)$ of prime ideals of A extending I . Since A is recursive, we may assume that the universe of A is ω . Let the operations of A be denoted by $+^A$ and $\cdot^A$ and assume that the additive identity $0^A = 0$ and the unity $1^A = 1$. Let I^s be the set of elements of I which have been enumerated into I by stage s . Then the recursive tree T is defined so that for any $\sigma = (x(0), \dots, x(n-1)) \in \{0, 1\}^n$, σ is in T if and only if the following conditions all hold.

- (i) For any $i, j, k < n$, if $i +^A j = k$, $x(i) = 1$ and $x(j) = 1$, then $x(k) = 1$.
- (ii) For any $i, j, k < n$, if $i \cdot^A j = k$ and $x(i) = 1$, then $x(k) = 1$.
- (iii) If $n > 1$, then $x(1) = 0$.
- (iv) For any $i, j, k < n$, if $i \cdot^A j = k$ and $x(k) = 1$, then $x(i) = 1$ or $x(j) = 1$.
- (v) For any $i < n$, if $i \in I^n$, then $x(i) = 1$.

Conditions (i), (ii) and (iii) ensure that any infinite path through T will represent an ideal of R . Condition (iv) ensures that any infinite path through T will represent a prime ideal. Condition (v) ensures that any infinite path through T will represent an extension of I . Note that we can modify this construction to define the class of all ideals of A which extend I by simply omitting condition (iv).

To define the class of maximal ideals of A which extend I , we note that any maximal ideal is prime and that an ideal J is maximal in A if and only if, for each $r \in A \setminus J$, the ideal $J(r)$ generated by $J \cup \{r\}$ equals A , which is if and only if $1 \in J(r)$, and we also note that $J(r) = \{i +^A r \cdot^A s : i \in J, s \in A\}$. Thus we let P be the Π_1^0 class representing the set of prime ideals of A extending I and define Q with $Q \subseteq P$ by

$$x \in Q \iff x \in P \ \& \ (\forall j) [x(j) = 0 \rightarrow (\exists i, r) (x(i) = 1 \ \& \ 1 = i +^A r j)].$$

Thus the set of all maximal ideals of A extending I is represented by the Π_2^0 class Q .

(b) The class representing all subgroups of the group $G = (\omega, +^G, 0, -^G)$ is defined as the set of all x such that all of the following hold.

(i) For any $i, j, k < n$, if $i +^G j = k$ and $x(i) = x(j) = 1$, then $x(k) = 1$,

(ii) For any $i, j, k < n$, if $i -^G j = k$ and $x(i) = x(j) = 1$, then $x(k) = 1$,

(iii) $x(0) = 1$.

For the maximal subgroups, we note that H is a maximal subgroup of G if and only if, for each $g \in G \setminus H$, the subgroup $H(g)$ generated by $H \cup \{g\}$ equals G and we also note that $H(g) = \{h +^G z \cdot g : h \in H, z \in \mathbb{Z}\}$. Thus we let P be the Π_1^0 class representing the set of subgroups of G and define $Q \subseteq P$ by $x \in Q$ if and only if

$$x \in P \ \& \ (\forall a, j) [x(j) = 0 \rightarrow (\exists i)(\exists z \in \mathbb{Z}) (x(i) = 1 \ \& \ a = i +^G z \cdot j)].$$

Thus the set of all maximal subgroups of G is represented by the Π_2^0 class Q .

(c) Observe that a Boolean algebra $\mathcal{B} = (B, \wedge^B, \vee^B, \neg^B, \leq^B, 0^B, 1^B)$ is also a commutative ring with unity where the ring operations are given by $a \cdot^B b = a \wedge^B b$ and $a +^B b = (a \vee^B b) \wedge^B \neg^B(a \wedge^B b)$. Furthermore, the Boolean ideals are identical with the ring ideals and a ring ideal is prime if and only if it is both prime and maximal as a Boolean ideal. Thus the result for r.e. Boolean algebras follows from part (a). $\square$

We can now derive a number of immediate corollaries from Theorem 5.1 by applying the results of Section 2. Below are a few examples.

Theorem 5.2

- (a) *Any r.e. presented recursive group which has a proper subgroup has a proper subgroup of r.e. degree.*
- (b) *If the set of prime ideals of a r.e. presented commutative ring with unity R is countably infinite, then R has a recursive prime ideal.*
- (c) *If a r.e. Boolean algebra B has only finitely many prime ideals, then every prime ideal of B is recursive.*
- (d) *If an r.e. presented ring R has a maximal ideal, then R has a prime ideal recursive in some Σ_1^1 set.*

Proof. A few remarks are necessary. For part (a), if G has a proper subgroup H , fix $h \in H$ and $g \notin H$ and consider the Π_1^0 class of subgroups of G which contain h and omit g . For part (d), combine Theorem 2.6 with Theorem 2.26. $\square$

The results given above for an r.e. Boolean algebra can be strengthened for a recursive Boolean algebra.

Theorem 5.3 *For any recursive Boolean algebra $\mathcal{B}$, the set of all ideals of $\mathcal{B}$ and the set $\text{Max}(R)$ of maximal ideals of R can always be represented as the Π_1^0 class of infinite paths through a recursive tree with no dead ends.*

Proof. Let $\mathcal{B} = (B, \wedge^B, \vee^B, \neg^B, \leq^B, 0^B, 1^B)$ be a recursive Boolean algebra with universe $B = \omega$. Then the recursive tree T such that $[T]$ represents the class of ideals of $\mathcal{B}$ is defined as follows. Let $\sigma = (x(0), \dots, x(n-1))$ be a finite sequence, let $a_1, \dots, a_k$ enumerate $\{i < n : x(i) = 1\}$, and let

$$j(\sigma) = a_1 \vee^B a_2 \vee^B \dots \vee^B a_k.$$

Then $\sigma \in T$ if and only if,

- (i) for any $a < n$, if $a \leq^B j(\sigma)$, then $\sigma(a) = 1$,
- (ii) $j_\sigma \neq 1$.

Clause (i) ensures that for any infinite path x through T , $I(x) = \{i : x(i) = 1\}$ will be closed under meet and join and also closed downward. Clause (ii) ensures that $1^B \notin I(x)$, so that $I(x)$ is an ideal of $\mathcal{B}$. It also ensures that any finite path $\sigma \in T$ has an infinite extension $x \in [T]$. To see this, simply let $I(\sigma) = \{a : a \leq j(\sigma)\}$ and let x be the characteristic function of $I(\sigma)$. The tree T_M such that $[T_M]$ represents the class of maximal ideals of $\mathcal{B}$ is defined by adding to the definition of T the clause:

(iii) For any $a, b < n$, if $b = \neg^B a$, then either $\sigma(a) = 1$ or $\sigma(b) = 1$.

Now suppose that $\sigma \in T_M$ and let $I(\sigma)$ be the ideal defined above. It follows from the Boolean Prime Ideal Theorem that $I(\sigma)$ can be extended to a maximal ideal M of $\mathcal{B}$ and it is clear that the characteristic function of M is an extension of σ and is in $[T_M]$, so that T_M likewise has no dead ends. $\square$

We obtain as a corollary of Theorem 5.3 together with Theorem 2.6 the following result from the folklore.

Corollary 5.4 *Any recursive Boolean algebra has a recursive maximal ideal.*

Next we turn to the other direction of our correspondences, that is, representing an arbitrary Π_1^0 class by the set of solutions to one of our problems. We say that a class Q *weakly represents* the class P if there is a recursive function φ such that for each $x \in Q$, $\varphi(x) \in P$ and $\varphi(x) \leq_T x$ and there is a recursive functional ψ such that for all $y \in Q$, $\psi(y) \in P$ and $\psi(y) \equiv_T y$.

Theorem 5.5

- (a) *Any r.b. Π_1^0 class P can be represented by the set of prime ideals of an r.e. Boolean algebra.*
- (b) *For any pair of disjoint r.e. sets, the r.b. Π_1^0 class $S(A, B)$ can be weakly represented by the set of prime ideals and by the set of maximal ideals of some r.e. commutative ring R with identity.*

Proof.

(a) We give two proofs which demonstrate the connection between Boolean algebras, logical theories and Π_1^0 classes. Let the r.b. Π_1^0 class $P = [T]$ be given. For the first proof, recall from Theorem 3.3 the language $\mathcal{L} = \{A_i : i < \omega\}$ and the theory $\Gamma = \Gamma(T)$ of that language such

that the class $P(\Gamma)$ of complete consistent extensions of Γ represents P . Let $\mathcal{B}(\mathcal{L})$ be the Lindenbaum algebra for $\mathcal{L}$. Then the theory $\Gamma(T)$ defined in Theorem 3.3 provides an r.e. ideal $\Gamma(T)^*$ of $\mathcal{B}(\mathcal{L})$, so that the class of prime ideals of $\mathcal{B}(\mathcal{L})/\Gamma(T)^*$ is in one-to-one effective correspondence with the class of complete consistent extensions of $\Gamma(T)$. The result now follows.

The alternate proof for (a) takes advantage of the topology on $\{0, 1\}^\omega$. The underlying recursive Boolean algebra here is simply the collection $\mathcal{B}(\{0, 1\}^\omega)$ of clopen subsets of $\{0, 1\}^\omega$, each of which is a finite union of intervals and can thus be represented by a finite string. For any Π_1^0 class P , we represent P by the Boolean algebra of relatively clopen subsets of P . That is, we define the r.e. ideal $I(P)$ of the $\mathcal{B}(\{0, 1\}^\omega)$ to be the family of clopen sets which intersect P . It is then easy to see that the class of prime ideals in $\mathcal{B}(\{0, 1\}^\omega)/I(P)$ is in effective one-to-one correspondence with P .

(b) We give the proof of Friedman-Simpson-Smith [54]. Let the infinite disjoint r.e. sets A, B be given. The construction begins with the underlying ring $R = \mathbb{Q}[x_n : n \in \omega]$ (the ring of polynomials with rational coefficients in infinitely many variables). Let $A = \{f(n) : n \in \omega\}$ and $B = \{g(n) : n \in \omega\}$ and let I be the ideal generated by $\{x_{f(n)}^{n+1}, x_{g(n)}^{n+1} - 1, n \in \omega\}$. We claim that:

- (1) I is a proper recursive ideal,
- (2) the set of prime ideals of R/I represents $S(A, B)$, and
- (3) the set of maximal ideals of R/I represents $S(A, B)$.

To test whether a given $f \in R$ is in I , we first produce $f^* = f \pmod{I}$ by repeating the following reduction procedure. For any factor x_m^{k+1} occurring in a term of f , enumerate the finite sets $F = \{f(0), \dots, f(k)\}$ and $G = \{g(0), \dots, g(k)\}$ and determine whether $m \in F \cup G$. If $m = f(i) \in F$, then replace x_m^{k+1} with 0 (since $x_m^{i+1} \in I$); if $m = g(i) \in G$, then replace x_m^{k+1} with x_m^{k-i} (since $x_m^{i+1} - 1 \in I$). Each step in this process reduces the degree of some term and thus the process must terminate in f^* after a finite number of steps. Then $f \in I$ if and only if this $f^* = 0$. It follows that $1 \notin I$, so that I is proper.

For any set $C \in S(A, B)$, let M_C be the ideal generated by the set of all $\{x_m : m \in C\} \cup \{x_n - 1 : x \notin C\}$. It is easy to see that, using the reduction procedure described in the previous paragraph, any polynomial will be equivalent to some $q \in \mathbb{Q}$. Thus M_C is a maximal ideal and $M_C \equiv_T C$.

Now an ideal J is said to be *radical* if $a^n \in I$ for any n implies that $a \in I$. It is clear that any prime ideal is radical and it is easy to check that in fact any maximal ideal is radical. Suppose that J is a radical ideal of R which extends I . Then it follows that $x_{f(n)} \in J$ and that $x_{g(n)} \notin J$ for all n . Thus we can define the weak representation of $S(A, B)$ by a class of ideals, simply letting $\varphi(J) = C$, where $m \in C \Leftrightarrow x_m \in J$. Clearly $C \leq_T J$ in this case. $\square$

The representation Theorem 5.5 has, as usual, a number of immediate corollaries.

First we briefly discuss the significance of isolated and limit points and of thin and minimal Π_1^0 classes in the context of Boolean algebras. Let $\mathcal{B}$ be an r.e. Boolean algebra and let $P(\mathcal{B})$ be the Π_1^0 class of prime ideals of $\mathcal{B}$. Then for any r.e. ideal J of $\mathcal{B}$, the set $P(J)$ of prime ideals of $\mathcal{B}$ which extend J is a Π_1^0 subclass of $P(\mathcal{B})$. Thus if $P(\mathcal{B})$ is thin, then $P(J)$ is relatively clopen in $P(\mathcal{B})$, so that J is a finitely generated ideal. (This follows by an argument similar to that given before Theorem 3.4 above for theories). An r.e. Boolean algebra $\mathcal{B}$ with this property (that every r.e. ideal of $\mathcal{B}$ is finitely generated) is said to be a *thin* r.e. Boolean algebra. If $P(\mathcal{B})$ is minimal, then either $P(J)$ is finite or else $P(\mathcal{B}) \setminus P(J)$ is finite. If the prime ideal J is isolated in $P(\mathcal{B})$, then $P(J)$ is a relatively clopen subset of $P(\Gamma)$, so that J is a finitely generated ideal. For more information, see [17] and [76]. We also note that Downey and Jockusch [40] recently constructed a rank one r.e. Boolean algebra which is not isomorphic to any recursive Boolean algebra (Feiner [48] had constructed one of infinite rank).

Theorem 5.6

- (i) (a) *There is an r.e. Boolean algebra $\mathcal{B}$ which has no recursive prime ideal.*
- (b) *There is a recursive commutative ring with unity which is not a field and which has no recursive prime ideal.*
- (ii) *There is an r.e. Boolean algebra $\mathcal{B}$ such that any two distinct prime ideals of $\mathcal{B}$ are Turing incomparable, where distinct means having infinite symmetric difference.*
- (iii) *If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} \leq_T \mathbf{0}'$, then there exists an r.e. Boolean algebra B such that B has a prime ideal of degree $\mathbf{a}$ but has no recursive ideal.*

- (iv) *There is an r.e. Boolean algebra $\mathcal{B}$ such that if $\mathbf{a}$ is the degree of any prime ideal of $\mathcal{B}$ and $\mathbf{b}$ is an r.e. degree with $\mathbf{a} \leq_T \mathbf{b}$, then $\mathbf{b} \equiv_T \mathbf{0}'$.*
- (v) *If $\mathbf{c}$ is any r.e. degree, then there exists an r.e. Boolean algebra $\mathcal{B}$ such that the set of r.e. degrees which contain prime ideals of $\mathcal{B}$ equals the set of r.e. degrees $\geq_T \mathbf{c}$.*
- (vi) *There is an r.e. Boolean algebra $\mathcal{B}$ such that if I is a prime ideal of $\mathcal{B}$ with $I \leq_T \mathbf{0}'$, then there exists a nonrecursive r.e. set A such that $A <_T I$.*
- (vii) *For any degree $\mathbf{a}$ such that either $\mathbf{a} \leq_T \mathbf{0}'$ or $\mathbf{0}' \leq_T \mathbf{a} \leq_T \mathbf{0}''$, there exists an r.e. Boolean algebra $\mathcal{B}$ with a prime ideal I of degree $\mathbf{a}$ such that I is the unique non-recursive prime ideal of $\mathcal{B}$ and such that any other prime ideal of $\mathcal{B}$ is finitely generated.*
- (viii) *There is a thin r.e. Boolean algebra $\mathcal{B}$ with a prime ideal I of degree $\mathbf{0}'$ such that I is the unique non-recursive prime ideal of $\mathcal{B}$, such that any other prime ideal of $\mathcal{B}$ is finitely generated, and such that for any r.e. ideal J of $\mathcal{B}$, either there are only finitely many prime ideals of $\mathcal{B}$ extending J or else all but finitely many of the prime ideals of $\mathcal{B}$ extend J .*

We next consider the special case of Π_1^0 classes with no dead ends. This theorem follows immediately from Theorem 3.5 by considering the Lindenbaum algebra in the same manner that Theorem 5.5 followed from Theorem 5.3.

Theorem 5.7 *Let T be a highly recursive tree with no dead ends and let $P = [T]$. Then there is a recursive Boolean algebra $\mathcal{B}$ and an effective one-to-one correspondence between the prime ideals of $\mathcal{B}$ and P .*

We can now apply the results from Section 2 to obtain the following.

Theorem 5.8

- (i) *For any degree $\mathbf{a} \leq_T \mathbf{0}'$, there exists a recursive Boolean algebra $\mathcal{B}$ with a prime ideal I of degree $\mathbf{a}$ such that I is the unique non-recursive prime ideal of $\mathcal{B}$ and such that any other prime ideal of $\mathcal{B}$ is finitely generated.*

- (ii) *There is a thin recursive Boolean algebra $\mathcal{B}$ with a unique non-recursive prime ideal I , such that any other prime ideal of $\mathcal{B}$ is finitely generated, and such that for any r.e. ideal J of $\mathcal{B}$, either there are only finitely many prime ideals of $\mathcal{B}$ extending J or else all but finitely many of the prime ideals of $\mathcal{B}$ extend J .*

6 Recursive graphs

In this section we consider a number of recursive combinatorial problems associated with recursive graphs. These include the coloring problem, the Hamiltonian circuit problem, the vertex partition problem, and various matching problems.

We begin by giving a list of the problems and the required definitions together with some of the history of each problem. Next we explain (in varying detail) how to prove that the set of solutions to any such problem can be represented by a recursively bounded Π_1^0 class. Then we apply the results of Section 2 to obtain corollaries which apply to the set of solutions of any such problem. Conversely we also consider for each problem, whether the set of solutions to such a problem can represent any r.b. Π_1^0 class. In each case, we show that the set of solutions to such a problem can represent the class of separating sets of any two disjoint r.e. sets. Then we apply the results of Section 2 to obtain corollaries which give the existence of “pathological” problems of each type.

(1) *Graph-coloring problems.*

A countable infinite graph $G = (V, E)$ consists of a subset V of the natural numbers called *vertices* together with a symmetric subset E of $V \times V$, called the *edges*. G is said to be recursive if the sets V and E are recursive. We say that vertices u, v are joined by an edge (u, v) . The *degree* of a vertex u of G is the cardinality of the set of vertices joined to u . A *k -coloring* of the graph G is a map g from V into $\{1, 2, \dots, k\}$ such that $g(u) \neq g(v)$ whenever $(u, v) \in E$. The *k -coloring problem* for a graph G is to determine whether G has any k -colorings. The set of solutions to this problem is the set of k -colorings of G . We make the convention that, unless stated otherwise, the graphs we shall discuss are assumed to be connected, have no loops or multiple edges, and have the property that each vertex v of G is of finite degree.

The graph coloring problem has been studied in combinatorics for over a century. Two classical results for finite graphs are Brooks' Theorem [12] that every graph with all vertices of degree $\leq k$ and with no $(k+1)$ -cliques is k -colorable, and the Four Color Theorem of Appel and Haken [2, 3, 4] that every planar graph is 4-colorable. These results are easily extended to infinite graphs by a compactness argument. A natural question is whether such results can be effectivized. The answer to this question is yes for Brooks' Theorem, that is, Schmerl showed in [135] that every recursive graph with all vertices of degree $\leq k$ and with no $(k+1)$ -cliques has a recursive k -coloring. On the other hand, the Four Color Theorem cannot be effectivized. Bean constructed in [7] a 3-colorable, recursive, planar connected graph which has no recursive k -coloring for any k .

A recursive graph $G = (V, E)$ is said to be *highly recursive* if there is a partial recursive function $f : V \rightarrow \omega$ such that, for each $v \in V$, $f(v)$ is the degree of v . Highly recursive graphs are of interest for several reasons. One reason is the result of Bean [7] that any highly recursive k -colorable graph has a recursive $2k$ -coloring, in contrast to the result cited above for arbitrary recursive graphs. This result was improved by Schmerl [134] from $2k$ to $2k-1$, who also showed that $2k-1$ is the best possible result. It follows from the work of Bean and Schmerl that every highly recursive planar graph has a recursive 6-coloring. This result has recently been improved by Carstens [13] from 6 to 5, but the highly recursive four color problem remains open. Bean showed in [7] that the set of k -colorings of a highly recursive graph is always a recursively bounded Π_1^0 class. Conversely, Remmel [129] showed that every r.b. Π_1^0 class can actually be strongly represented by a highly recursive k -coloring problem.

The problem of feasible graphs and colorings has been studied by Cenzer and Remmel in [22].

(2) Matching problems.

A recursive society $S = (B, G, K)$ consists of disjoint recursive sets B , the set of boys, and G , the set of girls, and a recursive binary relation $K \subseteq B \times G$. Here $K(b, g)$ means b knows g . The solutions in this case are the set of *marriages*, that is, one-to-one maps $f : B \rightarrow G$ such that $K(b, f(b))$ holds for all b . For any subset B' of B , let $K(B') = \{g : (\exists b \in B') K(b, g)\}$. Marshall Hall [63] extended the classical Philip Hall Theorem to infinite societies and proved that, for any countable society $S = (B, G, K)$, if every boy knows only

finitely many girls and, for any finite subset $B' \subseteq B$, $|B'| \leq |K(B')|$, then there is a marriage for S . We say that a recursive society $S = (B, G, K)$ is *highly recursive* if there is a partial recursive function $k : B \rightarrow \omega$ such that, for each $b \in B$, $k(b)$ equals the cardinality of $K(b)$. We say that S is *symmetrically highly recursive* if there is also a partial recursive function $\bar{k}$ such that, for each $g \in G$, $\bar{k}(g)$ is the cardinality of the set of boys which know g .

The problems which we consider are:

- (i) The general problem of finding a marriage in a highly recursive society S ,
- (ii) the surjective matching problem, that is, finding a marriage $f : B \rightarrow G$ which is both one-to-one and onto in a symmetrically highly recursive society S , and
- (iii) the surjective matching problem, where each person knows at most two other people in a symmetrically highly recursive society S .

Problems (i) and (ii) were analyzed by Manaster and Rosenstein in [100, 101], who showed that the set of marriages in case (i) and (ii) is always an r.b. Π_1^0 class, but does not always contain a recursive element. Moreover, Manaster, Rosenstein showed that in case (ii), the set of surjective marriages can represent an arbitrary r.b. Π_1^0 class. We note that problem (iii) contains a recursive version of Banach's strengthening of the Schroder-Bernstein theorem, which was shown to be noneffective by Remmel [128]. That is, suppose we take one-to-one recursive functions with recursive ranges $f : B \rightarrow G$ and $g : G \rightarrow B$ where B and G are recursive sets. Then we can form a highly recursive society $S = (B, G, K)$, where $K(x, y)$ holds if and only if $f(x) = y$ or $g(y) = x$. For such a society S , the only surjective marriages h arise from some partition $B = B_1 \cup B_2$, where $h = f[B_1 \cup g^{-1}[B_2]$, and the existence of such marriages are guaranteed by Banach's result. (See [128] for details.) It was shown by Remmel in [129] that the set of surjective marriages in case (iii) cannot represent an arbitrary r.b. Π_1^0 class in contrast to the Manaster-Rosenstein result for case (ii).

(3) The Vertex Partition Problem.

This problem, posed by S. Ulam, is to show that for each partition of the vertex set V of a graph $G = (V, E)$ into sets of uniformly bounded cardinality,

there is at least one set of the partition which is adjacent to m (or more) other sets of the partition. Here we say that two sets S_1 and S_2 are adjacent if there exist vertices $v_1 \in S_1$ and $v_2 \in S_2$ such that $(v_1, v_2) \in E$. The partition number m of a graph G is the least number m for which the statement is true. The vertex partition problem was studied by Cenzer and E. Howorka [18], who computed the vertex partition numbers of various well-known graphs, including the m -regular trees T_m and the planar mosaic graphs M_3 , M_4 and M_6 . The tree T_m may be viewed as $\{1, 2, \dots, m\}^*$. The graphs M_3 , M_4 and M_6 may be viewed as tilings of the plane by regular hexagons, squares and equilateral triangles. In each case, the partition number of the graph turns out to be the degree of the graph. In this situation, the Π_1^0 class arises from the dual problem. That is, given the graph G and numbers k and m , to find a k -partition P of the graph such that no set has m neighbors. Here a k -partition is a partition of V into sets of cardinality $\leq k$. The solution to such a problem may be represented as a function f from $V \times V$ into $\{0, 1\}$ which is to be the characteristic function of the equivalence relation with equivalence classes being the sets of the partition.

(4) *The Hamiltonian Circuit Problem.*

Let $G = (V, E)$ be a countably infinite graph. Two vertices u, v of G are *adjacent* if $(u, v) \in E$ and two edges (u_1, v_1) and (u_2, v_2) are *adjacent* if either $v_1 = u_2$ or $u_1 = v_2$. A *one-way* (respectively *two-way*) *Hamiltonian circuit* (or *Hamiltonian path*) for G is a one-to-one correspondence f between the natural numbers ω (resp. the integers $\mathbb{Z}$) and V such that consecutive vertices are adjacent, i.e., $(f(i), f(i+1)) \in E$ for all i . The dual concepts are the *one-way* (respectively *two-way*) *Euler path*, which is a one-to-one correspondence between the natural numbers ω (resp. the integers $\mathbb{Z}$) and E such that consecutive edges are adjacent. For each of these four notions, let us also define the associated notion of being such a path for a subgraph. That is, a one-way Hamiltonian sub-path for G will be a one-to-one embedding of the natural numbers into V such that consecutive vertices are adjacent. The other three definitions are similar.

In each case, the problem here is whether a given graph has such a path. We will focus on the sub-path problems.

Now we consider the correspondence between the set of solutions to a recursive combinatorial problem and the set of infinite paths through a highly recursive tree.

In all of the highly recursive graph problems above (except for the Hamiltonian and Euler paths), it is easy to show that the set of solutions to the problem is an r.b. Π_1^0 class. For the (full) Hamiltonian and Euler path problems, the natural representation of the set of solutions is not necessarily a closed set and the representation as a Π_1^0 class is not necessarily bounded.

We will give a formal definition below of the Π_1^0 class corresponding to the vertex partition problem since this is a new result. We will also give a proof for the Hamiltonian and Euler sub-paths. First, we give an informal description of how to obtain the Π_1^0 class. The idea is essentially the same in each case, so we describe it in the case of graph colorings and then tell how to modify the idea for the other problems. Suppose we start with a recursive graph $G = (V, E)$ where V is a subset of ω . Let $v_0 < v_1 < \dots$ be an increasing enumeration of V . We start with the full k -ary tree T_k , the set of all finite sequences whose entries are less than k . We think of each vertex v_i as corresponding to level i and each infinite path $\pi = (\pi_0, \pi_1, \dots)$ through T_k as coding the coloring which colors the vertex v_i with color π_i . Thus the infinite paths through T_k represent the possible ways of coloring the vertices of G . Finally, we must prune the tree T_k to a tree T_G which depends on G by allowing the node $(\pi_0, \dots, \pi_k)$ if and only if there is no edge $(v_i, v_j) \in E$ with $i, j \leq k$ and $\pi_i = \pi_j$. That is, we simply eliminate all nodes which correspond to illegal colorings of G . It then follows that since G is recursive, we can effectively check if $(\pi_0, \dots, \pi_k) \in T_G$, so that T_G will be a highly recursive tree and the set of infinite paths through T_G corresponds precisely to the set of k -colorings of G .

The same idea shows that all our other sets of solutions to the various recursively presented graph problems P correspond to r.b. Π_1^0 classes. That is, to construct a highly recursive tree T_P for P , we

- (a) let each level of the tree correspond to a vertex (or edge) of the graph,
- (b) let the branching at each level correspond to the possible choices for that element, e.g., which color it is given, which boy or girl it is married to, and
- (c) prune the fully branching tree to eliminate the nodes which correspond to illegal choices.

In each case, the fact that a particular path through the fully branching tree fails to be legal is for a finite reason which can be effectively checked from

the recursive presentation. This ensures that the pruned tree is highly recursive and that the set of infinite paths through the highly recursive pruned tree corresponds precisely to the legitimate solutions to the combinatorial problem. We leave the details in most of the cases above to the reader.

Theorem 6.1 *For each of the following problems, the set of solutions can be represented as a Π_1^0 class. The class is recursively bounded in cases (1), (2) and (3) and, if the graph is highly recursive, in cases (2), (3), (6) (a) and (b). The class is bounded in case (2) if each boy knows only finitely many girls and, in case (3) if each boy knows only finitely many girls and also each girl knows only finitely many boys. The classes are bounded in cases (6) (a) and (b) if each vertex has finite degree.*

- (1) *The k -colorings of a recursive graph.*
- (2) *The marriages in a recursive society.*
- (3) *The surjective matching problem in a recursive society.*
- (4) *The surjective matching problem, where each person knows at most two other people in a symmetrically highly recursive society S .*
- (5) *The k -partitions of a recursive graph such that no set in the partition is adjacent to m other sets.*
- (6) (a) *The one-way Hamiltonian (Euler) sub-paths starting from a fixed vertex in a recursive graph.*
- (b) *The two-way Hamiltonian (Euler) sub-paths through a fixed vertex in a recursive graph.*
- (c) *The one-way Hamiltonian (Euler) paths starting from a fixed vertex in a recursive graph.*
- (d) *The two-way Euler paths through a fixed vertex in a recursive graph.*

Proof. Part (1) is due to Bean [7] and parts (2), (3) and (4) are due to Manaster and Rosenstein [100, 101]. Note that in (2) the solution is simply a function from B into G , whereas in (3) the solution is represented as a pair of functions, one from B into G and one from G into B , which are inverses of each other.

(5) Let $G = (V, E)$ be a highly recursive graph and let k, m be positive integers. Let C be the set of k -partitions of V such that no set in the partition is adjacent to m other sets. As indicated above, we may represent a partition by the characteristic function f of the corresponding equivalence relation. Let us assume that $V = \omega$ for simplicity and let C be the class of all such functions for which there is no set in the partition represented by f which has m neighbors. Now a function $f \in \{0, 1\}^\omega$ will be in the class C if it satisfies the following conditions:

- (i) $(\forall u) [f(u, u) = 1]$.
- (ii) $(\forall u, v) [f(u, v) = f(v, u)]$.
- (iii) $(\forall u, v, w) [f(u, v) = f(v, w) = 1 \rightarrow f(u, w) = 1]$.
- (iv) $(\forall u_1, u_2, \dots, u_{k+1}) (\exists i, j \leq k+1) [f(u_i, u_j) = 0]$.
- (v) $(\forall u_1, v_1, u_2, v_2, \dots, u_m, v_m)$

$$[(\forall i, j \leq m) [f(u_i, u_j) = 1] \ \& \ (\forall i \leq m) [E(u_i, v_i)]]$$

$$\rightarrow (\exists i, j \leq m) [f(v_i, v_j) = 1]]$$
.

The first three clauses are the requirement that f is the characteristic function of an equivalence relation. The fourth clause is the requirement that each set in the corresponding partition has cardinality $\leq k$ and the final clause is the requirement that no set in the partition is adjacent to m other sets.

(6) (a) Let the recursive graph $G = (V, E)$ with fixed vertex v_0 be given. Then a one-way Hamiltonian (Euler) sub-path is a function f from ω into V with $f(0) = v_0$ such that $(f(n), f(n+1)) \in E$ for all n and such that, for the Hamiltonian path, $m \neq n$ implies that $f(m) \neq f(n)$ and, for the Euler path, $m \neq n$ implies that the edges $(f(m), f(m+1))$ and $(f(n), f(n+1))$ are different. In each case, this clearly defines a Π_1^0 class P . If each vertex v has finite degree, then there is a function g such that all vertices joined to vertex v are $\leq g(v)$. It follows that we can compute a bound $h(m)$ for the possible value of $f(m)$ by letting $h(0) = v_0$ and in general $h(m+1) = \sup\{g(v) : v \leq h(m)\}$. This shows that P is bounded. If G is highly recursive, then the function g may be taken to be recursive, so that P is recursively bounded.

(b) Again let the recursive graph $G = (V, E)$ with fixed vertex v_0 be given. Then a two-way Hamiltonian (Euler) sub-path

$$\dots, \pi(-1), \pi(0) = v_0, \pi(1), \dots$$

can be represented as a function f from ω into V with $f(0) = v_0$ such that $(v_0, f(1)) \in E$, such that $(f(n), f(n+2)) \in E$ for all n and such that, for the Hamiltonian path, the function f is one-to-one, and, for the Euler path, no edge occurs twice in the list

$$\dots, (f(3), f(1)), (f(1), f(0)), ((f(0), f(2)), (f(2), f(4)), \dots$$

It follows as in (a) that the class P of two-way Hamiltonian (Euler) sub-paths is a Π_1^0 class, is bounded if each vertex of G has finite degree, and is r.b. if G is highly recursive.

(c) We first give the proof for one-way Hamiltonian paths. Recall that $V = \omega$ and represent a one-way Hamiltonian path

$$\pi = (\pi(0) = v_0, \pi(1), \pi(2), \dots)$$

by a function f such that $f(2n) = \pi(n)$ and $f(2v+1) = n$ such that $v = \pi(n)$. This is clearly a one-to-one degree-preserving correspondence between the one-way Hamiltonian paths of G and the Π_1^0 class P . Then the Π_1^0 class P of solutions is the set of functions f such that $f(0) = v_0$, such that $(f(2n), f(2n+2)) \in E$ for all n , and such that, for all v and n , $f(2n) = v$ if and only if $f(2v+1) = n$. For the one-way Euler paths π , we take $f(2n) = \pi(n)$ and let $f(2[u, v]+1) = n+1$ such that $\pi(n) = u$ and $\pi(n+1) = v$ if $(u, v) \in E$ and otherwise $f(2[u, v]+1) = 0$. In either case, the assumption that G is highly recursive does not necessarily imply that P is even bounded.

(d) Represent a two-way Hamiltonian path by a function f so that the path is given by

$$\dots, f(4), f(1), f(0) = v_0, f(3), f(6), \dots$$

and such that $f(3v+2) = n$ such that $n \not\equiv 2 \pmod{3}$ and $f(n) = v$. Represent a two-way Euler path π again by a function f so that

$$\pi = \dots, f(4), f(1), f(0) = v_0, f(3), f(6), \dots$$

and now such that $f(3[u, v]+2) = n$ such that $n \not\equiv 2 \pmod{3}$ and $f(n) = u$ and $f(n+3) = v$. $\square$

In those cases of Theorem 6.1 where we have established the fact that our set of solutions corresponds to an r.b. Π_1^0 class, we can derive a number of immediate corollaries from the results of Section 2. For example, applying Theorems 2.7, 2.10, 2.12 and 2.6, we see that the following is true.

Theorem 6.2 *For each recursive instance of problem (1) or (5) and each highly recursive instance of one of the problems (2), (3), (6) (a) or (b) of Theorem 6.1 the following hold.*

- (a) *If P has a solution s , then P has a solution s' in some r.e. degree.*
- (b) *If P has a solution s , then P has solutions s_1 and s_2 such that any function recursive in both s_1 and s_2 is recursive.*
- (c) *If there are only countably many solutions to P , then P has a recursive solution.*
- (d) *If there are only finitely many solutions to P , then each solution is recursive.*
- (e) *If P has a solution but no recursive solution, then for any countable sequence of nonzero degrees $\{\mathbf{a}_i\}$, P has continuum many solutions s which are mutually Turing incomparable and such that the degree of s is incomparable with each $\mathbf{a}_i$.*

For a recursive instance of problems (2), (3), (4), (6) (a) or (b) in which every boy, girl or vertex (as appropriate) is joined with only finitely many others, it follows from Theorem 2.6 that if P has a solution, then P has a solution recursive in $\mathbf{0}''$. In cases (c) and (d) of the Hamiltonian and Euler paths, we can only conclude, even for a highly recursive graph G , that G has a solution recursive in some Σ_1^1 set. Bean showed that if G is highly recursive and has an Euler paths, then G will actually have a recursive Euler path.

Finally, we should note that the correspondence between solutions to a specific instance P of one of our combinatorial problems and infinite paths through the tree T_P constructed as outlined above can also be viewed as a means of extending the proof of the finite version of the combinatorial theorem to a countable version of the combinatorial problem. That is, if we take the case of the Halls' theorem, we start with a countably infinite society

$S = (B, G, K)$ such that every boy knows only finitely many girls and such that every finite set B' of boys knows at least $|B'|$ girls, then we can construct what we might call the tree T_P of partial marriages as indicated above. Thus to prove (the infinite, Marshall Hall theorem) that S has a marriage, we need only show that there is an infinite path through T_P , which in turn follows from König's Lemma if we can prove that T_P is infinite. But T_P must be infinite, since if we restrict the society to a finite subset $S' = (B', G', K')$, where B' is any finite subset of B , G' is the set of girls known by B' , and $K' = K \cap (B' \times G')$, then S' has a marriage by the (finite) Philip Hall theorem. But this last fact easily implies that there is at least one node at every level of T_P , so that T_P is infinite. Such extension results thus explain why r.b. Π_1^0 classes arise naturally in recursive combinatorics.

Now we will consider the reverse direction of these correspondences. That is, suppose we are given a highly recursive tree. Is it possible to construct an instance of a specific recursive combinatorial problem so that the set of solutions of the problem correspond to the infinite paths through the tree? Formally, when we say that there is an effective one-to-one degree-preserving correspondence between the set $\mathcal{S}(P)$ of solutions to some recursively presented combinatorial problem P and the set $[T]$ of infinite paths through a highly recursive tree, we mean that we have a recursive functional Φ such that the map $\Phi(s) = \pi$ is a one-to-one degree preserving map from $\mathcal{S}(P)$ to $[T]$. Here a recursive functional acting on input s can be viewed as a recursive function φ with oracle s such that, for any n , $\pi(n) = \varphi^s(n)$.

There are several positive results showing that a combinatorial problem can strongly represent any highly recursive tree, including the two cited above, that is, the case of the surjective matching problem due to Manaster and Rosenstein in [100] and the case of the k -coloring problem due to Remmel [129]. The case of the Hamiltonian paths of a highly recursive graph was proved by Bean [8]. A Hamiltonian path for the graph $G = (V, E)$ is a function f mapping ω one-to-one and onto V such that $(f(n), f(n+1)) \in E$ for all n . In this case, however, it should be noted that the set of Hamiltonian paths of a highly recursive graph is not necessarily an r.b. Π_1^0 class. This can be demonstrated from the work of Harel [64], who showed that the problem of the existence of (one-way or two-way) Hamiltonian paths in a highly recursive graph is Σ_1^1 -complete and therefore not Π_1^1 . Now if the set of Hamiltonian paths through a highly recursive graph could always be represented as an r.b. Π_1^0 class, then, by Theorem 2.6 (c), every highly recursive

tree with a Hamiltonian path would have a Hamiltonian path recursive in $\mathbf{0}'$ and hence such a path is hyperarithmetical. If that were the case, then the Spector-Gandy theorem [66, p. 147] would show that the set of highly recursive graphs with Hamiltonian paths is Π_1^1 , contradicting Harel's result.

Negative results include the case of the surjective matching problem where each vertex has degree two which cannot even degree represent every r.b. Π_1^0 class which was proved by Remmel in [128].

Now the representation of an arbitrary r.b. Π_1^0 class by the set of k -colorings of a recursive graph G is not quite a one-to-one correspondence, since for any k -coloring of G , there are $k!$ permutations of the colors which yield $k!$ different colorings of the graph which are essentially the same as the given coloring. Thus in the correspondence, these $k!$ colorings are identified. However the problem of k -colorings of *planar* graphs presents a more serious difficulty. In fact, the problem of k -coloring an infinite planar graph, for $k \geq 5$, cannot even degree represent every r.b. Π_1^0 class, by the following argument. Since every planar graph $\mathcal{G}$ is 4-colorable, by the theorem of Appel and Haken [2, 3, 4], let f be such a coloring. Now there must be some color, say 0, such that infinitely many vertices, $v_0, v_1, \dots$ all satisfy $f(v_i) = 0$. Then we can change the color of any subset of $\{v_0, v_1, \dots\}$ to the 5-th color to get a different 5-coloring of $\mathcal{G}$. It follows that $\mathcal{G}$ has continuum many different colorings and of course this implies that there are infinitely many degrees among these colorings. But there are certainly r.b. Π_1^0 classes which are finite.

Next we consider a restricted version of the strong representation question. Recall that if A and B are infinite disjoint r.e. sets, then C is a separating set for A and B if $A \subseteq C$ and $B \cap C = \emptyset$. Then the class $\mathcal{S}(A, B) \subseteq \mathcal{P}(\omega)$ of separating sets for A and B is an r.b. Π_1^0 class.

The problem of strongly representing every r.b. Π_1^0 class of separating sets as the class of solutions to a some instance of a given recursive combinatorial problem is of interest for a number of reasons. It is an easy way to exhibit recursive combinatorial problems with no recursive solutions. It also has a connection with reverse mathematics. Reverse mathematics is the program of Harvey Friedman (and others) to answer the question: *What set existence theorems are needed to prove ordinary theorems of mathematics?* We observed above that König's Lemma is used to prove the existence of solutions to these infinite combinatorial problems. The reverse mathematics

is the following. If a combinatorial problem can strongly represent the r.b. Π_1^0 class of separating sets for any pair of disjoint infinite r.e. sets, then the existence of solutions to the combinatorial problem also implies König's Lemma in a certain subsystem RCA_0 of second-order arithmetic. This result is contained in Lemma 2.6 of Simpson [139]. Thus our next theorem has the corollary that, for each problem listed, the existence theorem for solutions to the problem implies König's Lemma. For some of the problems cited, Hirst gave a direct proof of this corollary in [67].

Theorem 6.3 *The following problems can strongly represent the r.b. Π_1^0 class of separating sets for any pair of disjoint infinite r.e. sets.*

- (1) *The problem of finding a k -coloring for a k -colorable highly recursive graph, for any $k \geq 3$.*
- (2) *The problem of finding a marriage in a highly recursive society.*
- (3) *The problem of finding a surjective marriage in a symmetrically highly recursive society.*
- (4) *The problem of finding a surjective marriage in a symmetrically highly recursive society where each person knows at most two other people.*
- (5) *The problem of finding a 2-partition of a highly recursive graph such that no set in the partition is adjacent to 3 other sets.*
- (6) *The problem of finding a (one-way or two-way) Hamiltonian path for a highly recursive graph.*

Proof. Fix a pair A and B of infinite disjoint r.e. sets and recursive enumerations $\{A^s\}_{s \in \omega}$ and $\{B^s\}_{s \in \omega}$ such that, for all s , $A^s, B^s \subseteq \{0, 1, \dots, s\}$ and there is at most one element of $A \cup B$ which comes into $A \cup B$ at stage s . For each of the problems above, the proof that it can strongly represent the class of separating sets of A and B is simple enough that in each case we shall only give a brief sketch of the basic idea and leave the details to the reader.

The k -coloring problem (1), the surjective matching problem (3) and the Hamiltonian path problem (6) are covered by the fact that they can strongly represent any r.b. Π_1^0 class.

(2) For each $i \in \omega$, we will specify a boy b_i and two girls $g_{0,i}$ and $g_{1,i}$ so that b_i knows both $g_{0,i}$ and $g_{1,i}$ and no other. Our highly recursive society $S = (B, G, K)$ will be such that $G = \{g_{0,i}, g_{1,i} : i \in \omega\}$ and $B = R \cup \{b_i : i \in \omega\}$, where $R = \{r_s : (A^s \cup B^s) - (A^{s-1} \cup B^{s-1}) \neq \emptyset\}$ is some infinite set of boys held in reserve. Again a marriage f for S will code a set C_f by specifying that $i \in C_f$ if and only if $f(b_i) = g_{1,i}$. We then determine who the boys in R know in stages in such a way that

- (a) if $i \in A$, then one boy in R knows $g_{1,i}$ and no others and no boy in R knows $g_{0,i}$,
- (b) if $i \in B$, then one boy in R knows $g_{0,i}$ and no others and no boy in R knows $g_{1,i}$,
- (c) if $i \notin A \cup B$, then no boy in R knows $g_{0,i}$ or $g_{1,i}$.

Then if i enters $A \cup B$ at stage s , we put $r_s \in B$ and we put $(r_s, g_{1,i})$ in K if $i \in A$ and $(r_s, g_{0,i})$ in K if $i \in B$. It is clear that this defines a highly recursive society S and that there is a one-to-one degree preserving correspondence between the marriages f for S and the separating sets C of A and B , given by mapping f to C_f .

(4) We first partition ω into a recursive sequence of infinite recursive sets $(G_0, B_0, G_1, B_1, \dots)$. For any fixed i , let $g_i^0 < g_i^1 < \dots$ and $b_i^0 < b_i^1 < \dots$ list the elements of G_i and B_i in increasing order. Our symmetrically highly recursive society $S = (B, G, K)$ will be thought of as a bipartite graph with $B = \cup_i B_i$ and $G = \cup_i G_i$. The idea is to construct a connected component of S with vertex set $G_i \cup B_i$ for each i . We construct the i -th component in stages, so that at stage s , we determine the edges out of g_i^s and b_i^s . We begin as if we are going to construct the two-way infinite chain in which b_i^0 is joined to g_i^0 and g_i^1 and such that, for each $n > 0$, b_i^{2n} is joined to g_i^{2n-2} and g_i^{2n} and b_i^{2n-1} is joined to g_i^{2n-1} and g_i^{2n+1} . See Figure 1.

Observe that there are exactly two possible surjective marriages f for such a component depending on whether $f(b_i^0) = g_i^0$ or $f(b_i^0) = g_i^1$. A marriage $f : B \rightarrow G$ for S will code a separating set C_f for A and B by letting $i \in C_f$ if and only if $f(b_i^0) = g_i^1$. Then it is easy to see that all we need to do to ensure that each marriage f of S corresponds to a separating set C_f for A and B is to construct the i -th component so that it is a one-way chain starting in B_i if $i \in A$, a one-way chain starting in A_i if $i \in B$, and the full two-way infinite

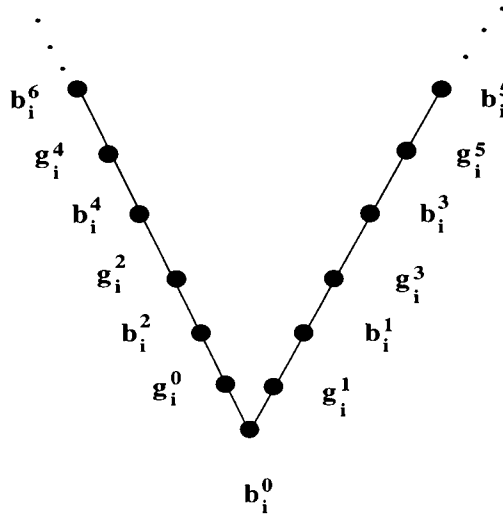


Figure 1: Generic component of the symmetric society

chain if $i \notin A \cup B$. Thus we build the chain until we see that $i \in A \cup B$ at some stage s . That is, at each stage t , we add b_i^k and g_i^k for $k \in \{2t, 2t + 1\}$ as pictured in Figure 1. Then if $i \in B^s$ omit b_i^{2n+1} and g_i^{2n+1} from the chain for all $n \geq s$ and if $i \in A^s$, then we omit b_i^{2n} and g_i^{2n} from the chain for all $n \geq s$.

(5) The graph (V, E) is defined as follows. We partition the vertex set $V = \omega$ into a recursive sequence of infinite recursive sets $(U_0, V_0, U_1, V_1, \dots)$. For each fixed i , let $u_{i,0} < u_{i,1} < u_{i,-1} < u_{i,2} < \dots$ and $v_{i,0} < v_{i,1} < v_{i,-1} < \dots$ list the elements of U_i and V_i in increasing order. The graph V will be the union of connected components $(U_i \cup V_i, E_i)$. For each i , E_i contains all of the edges $(u_{i,j}, u_{i,j+1})$ and $(v_{i,j}, v_{i,j+1})$ as well as the edges $(u_{i,j}, v_{i,j})$ and $(v_{i,j}, u_{i,j+1})$. See Figure 2.

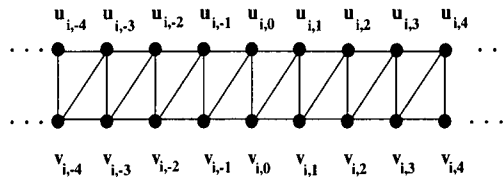


Figure 2: Graph for 2-partition problem

Observe that there are exactly two possible 2-partitions of the component $U_i \cup V_i$ such that no set has 3 neighbors, one being $\{\{u_{i,j}, v_{i,j}\} : j \in \mathbb{Z}\}$ and the other being $\{\{u_{i,j+1}, v_{i,j}\} : j \in \mathbb{Z}\}$. Such a partition P will code a separating set C_P for A and B by letting $i \in C_P$ if and only if $\{u_{i,1}, v_{i,1}\} \in P$. Thus we add the following edges to the graph. If i enters A at stage s , then we add the edge $(u_{i,s}, v_{i,s+1})$ which forces the set $\{u_{i,s}, v_{i,s-1}\}$ to be connected to $u_{i,s-1}, u_{i,s+1}, v_{i,s-2}, v_{i,s}$ and $v_{i,s+1}$ and hence have at least three neighbors. Thus in this case any legal partition P must include $\{u_{i,s}, v_{i,s}\}$ for all s and hence $i \in C_P$. If i enters B at stage s , then we add the edge $(v_{i,s}, u_{i,s+2})$ which forces the set $\{u_{i,s}, v_{i,s}\}$ to be connected to $u_{i,s-1}, u_{i,s+1}, u_{i,s+2}, v_{i,s-1}$ and $v_{i,s+1}$ and hence have at least three neighbors. Thus in this case any legal partition P must include $\{u_{i,s}, v_{i,s-1}\}$ for all s and hence $i \notin C_P$.

This completes the proof of Theorem 7.3. $\square$

Observe that in each of these problems, we have the notion of a recursive sub-problem determined by a recursive subset of the graph society, together with the relations determined by the whole problem. Likewise we have the notion of a solution to such a sub-problem, which provides a partial solution to the entire problem. Of course, the restriction of any (entire) solution to a sub-problem is a solution to the sub-problem. Let us say that a solution has a *limit point* if it is the limit of a sequence of distinct solutions (equivalently, if it belongs to the Cantor-Bendixson derivative of the class of solutions).

Theorem 6.4

(a) *For each one of the combinatorial problems listed above in Theorem 6.3:*

- (1) *There is a recursive instance P of the problem which has no recursive solution.*
- (2) *There is a recursive instance P of the problem such that any two distinct solutions of P are Turing incomparable, where distinct means differing in infinitely many places and also, for the coloring problem, that one solution cannot be obtained from the other by a permutation of the colors.*
- (3) *If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} \leq_T \mathbf{0}'$, then there is a recursive instance P of the problem such that P has a solution of degree $\mathbf{a}$ but has no recursive solution.*

- (b) For the k -coloring problem (1), the surjective matching problem (3) and the Hamiltonian path problem (6), the following also hold.
- (4) There is a recursive instance P of the problem such that if $\mathbf{a}$ is the degree of any solution of P and $\mathbf{b}$ is an r.e. degree with $\mathbf{a} \leq_T \mathbf{b}$, then $\mathbf{b} = \mathbf{0}'$.
 - (5) If $\mathbf{c}$ is any r.e. degree, then there exists a recursive instance P of the problem such that the set of r.e. degrees which contain solutions of P equals the set of r.e. degrees $\geq_T \mathbf{c}$.
 - (6) There is a recursive instance P of the problem such that if x is a solution of P with $x <_T \mathbf{0}'$, then there exists a nonrecursive r.e. set A such that $A <_T x$.
 - (7) For any degree $\mathbf{a} \leq_T \mathbf{0}'$, there exists a point x of degree $\mathbf{a}$ and a recursive instance P of the problem such that x is the unique non-recursive solution of P and is also the unique limit solution of P .
 - (8) There exists a recursive instance P of the problem such that:
 - (i) P has a unique non-recursive solution y which is also the unique limit solution and has degree $\mathbf{0}'$,
 - (ii) if R is any recursive sub-problem of P , and z is any recursive solution of F , then either (I) there are only finitely many solutions of P which extend z , or (II) all but finitely many solutions of P extend z .
 - (iii) if x is any recursive solution of P , then there is some finite sub-problem F of P such that any solution of P which agrees with x on F must equal x .

Proof.

(a) Parts (1) and (3) follow from the result of Shoenfield cited in Section 1.6, since A is itself a separating set for A and B . Part (2) follows from Theorem 2.31.

(b) Parts (4) and (5) follow from Theorem 2.9. Part (6) follows from the result of Kucera cited after Theorem 2.9.

(7) Let A be a set of degree $\mathbf{a}$ and let $B \equiv_T A$ be given by Theorem 2.16 (b). Thus there is a Π_1^0 class Q of sets such that $B \in Q \setminus D(Q)$. We

may assume without loss of generality that $D(Q) = \{B\}$ by restricting Q to a neighborhood U of B such that $D(Q) \cap U = \{B\}$. It follows that every other element of Q is isolated in Q and therefore recursive by Theorem 2.12. Now let the recursive problem P be given by Theorem 6.3 so that the set of solutions to P is represented by Q . Then the desired solution x is that which represents A .

(8) Let the set A be given by Theorem 2.20. Thus there is a minimal, thin Π_1^0 class Q such that $D(Q) = \{A\}$. Let the recursive problem P be given by Theorem 6.3 so that the set of solutions to P represents Q . Part (i) follows as in (7) above. For part (ii), note that the set of solutions of P which extend z represents a Π_1^0 subclass of Q and use the minimality of Q . For part (iii), use the fact that any recursive member of a thin Π_1^0 class must be isolated. $\square$

We can also translate index set results for graph colorings just like we did for logical theories. That is, we can effectively list all primitive recursive graphs $G_0, G_1, \dots$. It is easy to check that the reduction of Theorem 6.1 yields a recursive function p such that for each primitive recursive graph G_e , there is a primitive recursive k -ary tree $T_{p(e)}$ and an effective one-to-one degree preserving correspondence between the k -colorings of G_e and the Π_1^0 class $[T_{p(e)}]$. Vice versa, the correspondence due to Remmel [128] mentioned in Theorem 6.3 gives a recursive function q such that for any primitive recursive tree T_e , there is a primitive recursive graph $G_{q(e)}$ such that there is a one-to-one effective degree preserving correspondence between $[T_e]$ and the set of k -colorings of $G_{q(e)}$. We can then use these two correspondences and Theorem 2.50 to prove the following.

Theorem 6.5

- (a) $\{e : G_e \text{ has no } k\text{-colorings}\}$ is Σ_1^0 complete, and
 $\{e : G_e \text{ has a } k\text{-coloring}\}$ is Π_1^0 complete.
- (b) For any positive integer c ,
 $(\{e : G_e \text{ has } \leq k!c \text{ } k\text{-colorings}\}, \{e : G_e \text{ has } > k!c \text{ } k\text{-colorings}\})$
 is (Σ_2^0, Π_2^0) complete,
 $\{e : G_e \text{ has exactly } k!(c+1) \text{ } k\text{-colorings}\}$ is D_2^0 complete, and
 $\{e : G_e \text{ has exactly } k! \text{ } k\text{-colorings}\}$ is Π_2^0 complete.

- (c) $(\{e : G_e \text{ has } \geq \aleph_0 \text{ } k\text{-colorings}\}, \{e : G_e \text{ has } < \aleph_0 \text{ } k\text{-colorings}\})$
 is (Π_3^0, Σ_3^0) complete.
- (d) $(\{e : G_e \text{ has a rec. } k\text{-coloring}\}, \{e : G_e \text{ has no rec. } k\text{-coloring}\})$
 is (Σ_3^0, Π_3^0) complete.
- (e) For any positive integer c ,
 $(\{e : G_e \text{ has } \geq k!c \text{ rec. } k\text{-colorings}\},$
 $\{e : G_e \text{ has } < k!c \text{ rec. } k\text{-colorings}\})$
 is (Σ_3^0, Π_3^0) complete,
 $\{e : G_e \text{ has exactly } k!c \text{ rec. } k\text{-colorings}\}$ is D_3^0 complete.
 $(\{e : G_e \text{ has } < \aleph_0 \text{ rec. } k\text{-colorings}\},$
 $\{e : G_e \text{ has } = \aleph_0 \text{ rec. } k\text{-colorings}\})$
 is (Σ_4^0, Π_4^0) complete.

Similar translations can be done for the other problems that can strongly represent arbitrary r.b. bounded Π_1^0 classes. Also one can use the index sets for separating sets given in Section 2 to given index sets results for the problems which can represent an arbitrary class of separating sets. We give an example of this type of translation in the next section.

7 Recursive partial orderings

In this section we consider three problems associated with partially ordered sets (posets). Two of these are the dual problems of covering a poset with chains or with antichains. The third problem is the dimension problem, that is, expressing a poset as the intersection of linear orderings.

We proceed as in Section 6. We first describe the problems and show that the solution set to a recursive problem always forms an r.b. Π_1^0 class, and then apply the results of Section 2 to obtain corollaries which apply to the set of solutions of any such problem. We also consider for each problem, whether, conversely, the set of solutions to such a problem can represent any r.b. Π_1^0 class. For each problem, we show that the set of solutions to such a problem can represent the class of separating sets of any two disjoint r.e. sets and we apply the results of Section 2 to obtain corollaries which give the existence of “pathological” problems of each type.

(1) *Decomposition problems for posets.*

In this case, we start with a recursive poset $\mathcal{A} = (A, \leq^A)$, which consists of a recursive subset A of ω and a recursive ordering relation $\leq^A$. The *width* of $\mathcal{A}$ is the maximum cardinality of an antichain in $\mathcal{A}$ and the *height* of $\mathcal{A}$ is the maximum cardinality of a chain in $\mathcal{A}$.

(a) The first decomposition theorem we consider is Dilworth's theorem [34], which states that any poset $\mathcal{A}$ of width n can be covered by n chains. The problem here is to find such a covering of $\mathcal{A}$ by n chains and the set of solutions corresponds to the various coverings of $\mathcal{A}$ by n chains. The effective version of Dilworth's theorem has been analyzed by Kierstead in [77], where he showed that every recursive poset $\mathcal{A}$ of width n can be covered by $(5^n - 1)/4$ recursive chains, while for each $n \geq 2$, there are recursive posets of width n which cannot be covered by $4(n - 1)$ chains. See Kierstead's article [81] in this volume for details and more results.

Thus, the set of solutions of this problem for a recursive poset $\mathcal{A}$ can be represented as the set of maps $f : A \rightarrow \{1, 2, \dots, n\}$ such that $f^{-1}(\{i\}) = \{x \in A : f(x) = i\}$ is a chain for each i , which is clearly an r.b. Π_1^0 class.

(b) There is a natural dual to Dilworth's theorem which says that every poset of height n can be covered by n antichains. The problem again is to find such a covering. The effective version of the latter theorem was analyzed by Schmerl, who showed that every recursive poset of height n can be covered by $(n^2 + n)/2$ recursive antichains while for each $n \geq 2$, there is a recursive poset of height n which cannot be covered by $(n^2 + n)/2 - 1$ recursive antichains. Furthermore, Szemerédi and Trotter showed that there exist recursive partial orders of height n and recursive dimension 2 which still cannot be covered by $(n^2 + n)/2 - 1$ recursive antichains. These results are reported by Kierstead in [77].

The set of solutions for the dual problem for a recursive poset $(A, \leq^A)$ of height n can be represented as the set of maps $g : A \rightarrow \{1, 2, \dots, n\}$ such that $g^{-1}(\{i\})$ is an antichain for each i and is again an r.b. Π_1^0 class.

(2) *Dimension of posets problem.*

The poset $\mathcal{A} = (A, \mathcal{R})$ is defined to be n -dimensional if there are n linear orderings of A , $(A, L_1), \dots, (A, L_n)$, such that $R = L_1 \cap \dots \cap L_n$. The notion of the dimensionality of posets is due to Dushnik and Miller, who showed in [46] that a countable poset (A, R) is n -dimensional if and only if it can be

embedded as a subordering in the product ordering $\mathbb{Q}^n$, where $\mathbb{Q}$ is the set of rational numbers under the usual ordering. A (recursive) poset (A, R) has (recursive) dimension equal to d , for d finite, if there are d (recursive) linear orderings $(A, L_1), \dots, (A, L_d)$ such that $R = L_1 \cap \dots \cap L_d$, but there are not $d - 1$ (recursive) linear orderings $(A, L'_1), \dots, (A, L'_{d-1})$ such that $R = L'_1 \cap \dots \cap L'_{d-1}$. Kierstead, McNulty and Trotter have analyzed in [82], the recursive dimension of recursive posets and have shown that in general, the recursive dimension of a poset is not equal to its dimension.

Given a countable poset (A, R) with $A \subseteq \omega$, we can code a set of d linear orderings of A , $(A, L_1), \dots, (A, L_d)$ as follows. Let $a_0 < a_1 < \dots$ be an increasing enumeration of A . Then given d linear orderings of $\{a_0, \dots, a_{n-1}\}$, there clearly are $(n + 1)^d$ ways to extend the d linear orderings to d linear orderings on $\{a_0, \dots, a_n\}$. One can fix some effective enumeration of these extensions for each n , so that it then becomes possible to code each d -tuple of linear orderings by a function $f : A \rightarrow \omega$ where $f(a_n) \leq (n + 1)^d - 1$ for all n . Thus the set of solutions for the n -dimensionality problem of a recursive poset (A, R) can be represented as the set of all $f : A \rightarrow \omega$ such that f codes an n -tuple, $(A, L_1), \dots, (A, L_n)$, of linear orderings on A such that $R = L_1 \cap \dots \cap L_n$, which is an r.b. Π_1^0 class.

We state the first theorem and leave the details of the representation to the reader.

Theorem 7.1 *For each specific recursively presented instance of one of the poset problems P listed above, the set of solutions can be represented as an r.b. Π_1^0 class.*

As in Section 6, we can now derive a number of immediate corollaries from the results of Section 2. We state only a few of these and leave the rest to the reader. For example, the following is true.

Theorem 7.2

- (a) *If a recursive poset $\mathcal{A}$ has a covering by n chains, then $\mathcal{A}$ can be covered by n chains $C_1, \dots, C_n$ such that $C_1 \oplus \dots \oplus C_n$ has r.e. degree.*
- (b) *If $\mathcal{A} = (A, R)$ is a recursive poset such that the family of sets*

$$\{(A, L_1), (A, L_2), \dots, (A, L_n)\}$$

of n linear orderings such that $R = L_1 \cap L_2 \cap \dots \cap L_n$ is countably infinite, then $\mathcal{A}$ has recursive dimension $\leq n$.

- (c) *If a recursive poset $\mathcal{A}$ has a covering by n antichains, but has no covering by n recursive antichains, then for any countable sequence of nonzero degrees $\{\mathbf{a}_i\}$, $\mathcal{A}$ has a continuum of coverings $\{A_1, A_2, \dots, A_n\}$ by n antichains, which are pairwise Turing incomparable and such that the degree of $\{A_1, A_2, \dots, A_n\}$ is incomparable with each $\mathbf{a}_i$.*

Next we consider the reverse direction of this correspondence.

Theorem 7.3 *Each of the three problems described above can strongly represent the r.b. Π_1^0 class of separating sets for any pair of disjoint infinite r.e. sets.*

Proof. As in the proof of Theorem 6.3, fix a pair A and B of infinite disjoint r.e. sets and recursive enumerations $\{A^s\}_{s \in \omega}$ and $\{B^s\}_{s \in \omega}$ such that, for all s , $A^s, B^s \subseteq \{0, 1, \dots, s\}$ and there is at most one element of $A \cup B$ which comes into $A \cup B$ at stage s .

(1) The problem of covering a recursive poset of width k by k chains.

First consider the case $k = 2$. We begin with the poset $\mathcal{D}_0$ consisting of two one-way chains $\{a_{i,j} : i = 0, 1 \wedge j \in \omega\}$ and $\{b_{i,j} : i = 0, 1 \wedge j \in \omega\}$ where we have $a_{i,j} \leq a_{i,k}$ whenever $j < k$ and $a_{0,j} \leq a_{1,j}$ as well, and similarly for the $b_{i,j}$. The two chains are linked by having $a_{0,j} \leq b_{1,j}$ and similarly $b_{0,j} \leq a_{1,j}$. Let us call the posets $\{a_{0,i}, a_{1,i}, b_{0,i}, b_{1,i}\}$ the i -th block of the poset $\mathcal{D}_0$. The i -th block of $\mathcal{D}_0$ is pictured in Figure 3 (A).

Our final poset $\mathcal{D} = (D, \leq_D)$ will consist of the poset $\mathcal{D}_0$ together with an infinite recursive set E whose relations to the elements of $\mathcal{D}_0$ and among themselves is to be specified in stages. Now it is clear that a decomposition of this poset, up to renaming the chains, is completely determined by the choice, for each i , of either

- (a) putting $a_{0,i}$ and $a_{1,i}$ in one chain and $b_{0,i}$ and $b_{1,i}$ in the other, or
- (b) putting $a_{0,i}$ and $b_{1,i}$ in one chain and $a_{1,i}$ and $b_{0,i}$ in the other.

Thus we can think of a chain decomposition $f : D \rightarrow \{1, 2\}$ as coding up a set C_f , where $i \in C_f$ if and only if we use choice (b) for the i -th component, that is, if and only if $f(a_{0,i}) = f(b_{1,i})$. Now the idea is to define the relations between the remaining recursive set E so that we introduce an element e in the i -th component between $a_{0,i}$ and $a_{1,i}$ if $i \in B$, see Figure 3 (B). This

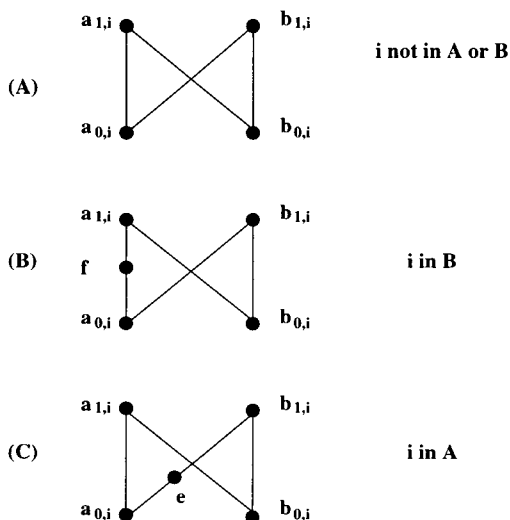


Figure 3: Blocks for width 2 poset

will force e , $a_{0,i}$ and $a_{1,i}$ to be in the same chain. We introduce an element f in the i -th component between $b_{0,i}$ and $a_{1,i}$ if $i \in A$, see Figure 3 (C). This will force f , $b_{0,i}$ and $a_{1,i}$ to be in the same chain. Finally we have no new element in the i -th component if $i \notin A \cup B$. It is not difficult to see that this can be accomplished so as to ensure that $\mathcal{D}$ is a recursive poset of width 2 and that such actions will ensure that the correspondence $f \rightarrow C_f$ will be a one-to-one degree preserving correspondence between the decompositions of $\mathcal{D}$ into two chains and the separating sets of $A \cup B$. We leave the details to the reader.

For the case where $k > 2$, one simply adds to the poset described a set of $k-2$ recursive infinite one-way chains, all of whose elements are incomparable with $\mathcal{D}$ and so that elements from different chains are also incomparable.

(2) The problem of covering a recursive poset of width k by k antichains.

Again we shall initially consider the case $k = 2$. The poset $\mathcal{D} = (D, \leq_D)$ will consist of two parts. The first part of the poset will consist of a recursive antichain $c_0, c_1, \dots$, and the second part will consist of two antichains $a_0, a_1, \dots$ and $b_0, b_1, \dots$ where $a_0 \leq b_0$ and, for each i , $a_{i+1} \leq b_i$ and $a_{i+1} \leq b_{i+1}$, see Figure 4.

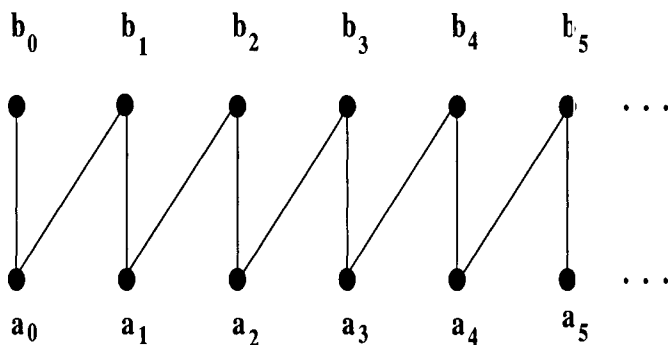
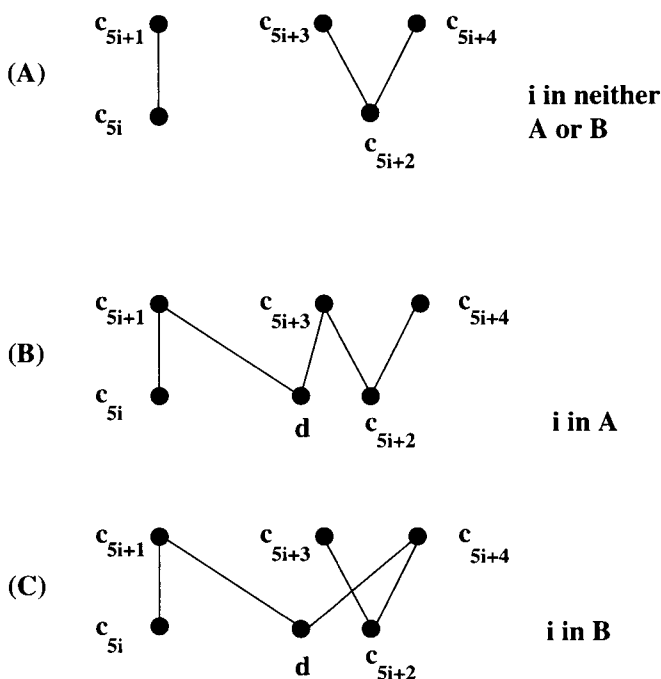


Figure 4: Height 2 poset

We will complete the partial ordering on $\mathcal{D}$ by specifying the relations between the two parts in stages. Clearly, up to renaming the antichains, there is a unique decomposition of the second part of the poset into two antichains. We think of a decomposition of $\mathcal{D}$ into two antichains as coding up a set C_f by specifying $i \in C_f$ if and only if f assigns c_i to the same antichain as the a 's. Then, for each i , we define c_i to be greater than a_s if $i \in A^{s+1} \setminus A^s$ and incomparable to a_s otherwise, and define c_i to be less than b_s if $i \in B^{s+1} \setminus B^s$ and incomparable to b_s otherwise. It is then easy to check that $\mathcal{D}$ is a recursive poset of height two and that, up to renaming the antichains, the correspondence $f \rightarrow C_f$ is a one-to-one degree preserving correspondence between decompositions of $\mathcal{P}$ into two antichains and separating sets of $A \cup B$. For the case where $k > 2$, one simply adds to the poset described a set of $k - 2$ recursive infinite antichains, all of whose elements are comparable with every element of $\mathcal{D}$ and so that elements from different antichains are also comparable.

(3) The problem of expressing a recursive poset $\mathcal{P} = (\mathcal{P}, \leq \mathcal{P})$ of dimension d as the intersection of d linear orderings.

We consider the case of two dimensional partial orderings. First we partition ω into two infinite recursive sets $C = \{c_0 < c_1 < \dots\}$ and $D = \{d_0 < d_1 < \dots\}$. For each i , we let $C_i = \{c_{5i}, c_{5i+1}, c_{5i+2}, c_{5i+3}, c_{5i+4}\}$. We shall define a recursive partial ordering $<_P$ on ω in stages. Given any two sets E and F , $E <_P F$ will denote that, for any $e \in E$ and $f \in F$, $e <_P f$. We start by defining $<_P$ so that $C_0 <_P C_1 <_P C_2 <_P \dots$. This means that if $<_1$ and $<_2$ are two linear orderings such that $<_1 \cap <_2 = <_P$, then the only

Figure 5: The i -th block of the dimension 2 poset

difference between $<_1$ and $<_2$ on C is how $<_1$ and $<_2$ order the elements within the blocks C_i . For each block C_i , $<_P$ is defined so that we have the Hasse diagram in Figure 5 (A).

It is then easy to check that, up to a permutation of the indices of the linear orderings $<_1$ and $<_2$, there are precisely two ways to define $<_1$ and $<_2$ on C_i so that $<_1 \cap <_2$ equals $<_P$ restricted to A_i , namely,

$$(I) \quad c_{5i} <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+3} <_1 c_{5i+4} \quad \text{and}$$

$$c_{5i+2} <_2 c_{5i+4} <_2 c_{5i+3} <_2 c_{5i} <_2 c_{5i+1}, \quad \text{or}$$

$$(II) \quad c_{5i} <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+4} <_1 c_{5i+3} \quad \text{and}$$

$$c_{5i+2} <_2 c_{5i+3} <_2 c_{5i+4} <_2 c_{5i} <_2 c_{5i+1}.$$

Note that the difference between (I) and (II) is that in the ordering where the elements c_{5i} , c_{5i+1} precede the elements c_{5i+2} , c_{5i+3} , c_{5i+4} , we have c_{5i+3} preceding c_{5i+4} in (I), while in (II) c_{5i+4} precedes c_{5i+3} .

We can thus use a pair of linear orderings $<_1$ and $<_2$ such that $<_1 \cap <_2 = <_P$ is defined within the blocks C_i to code a set $S(<_1, <_2) \subseteq \omega$ by declaring $i \in S$ if and only if $<_1$ and $<_2$ are of type (I) on C_i .

The key to our ability to code up a tree of separating sets for a pair of disjoint r.e. sets A and B is the following. If we add an element d to the Hasse diagram as pictured in Figure 5 (B), then only linear orderings $<_1$ and $<_2$ of type (I) can be extended to $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$ and if we add an element d to the Hasse diagram as pictured in Figure 5 (C), then only linear orderings $<_1$ and $<_2$ of type (II) can be extended to $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$.

That is, it is easy to check that, up to a permutation of indices there is only one way to define linear orderings $<_1$ and $<_2$ on $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$ if $<_P$ has the Hasse diagram as pictured in Figure 5 (B), namely

$$(I') \quad c_{5i} <_1 d <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+3} <_1 c_{5i+4} \quad \text{and} \\ c_{5i+2} <_2 c_{5i+4} <_2 d <_2 c_{5i+3} <_2 c_{5i} <_2 c_{5i+1}.$$

Similarly, up to a permutation of indices, there is only one way to define linear orderings $<_1$ and $<_2$ on $C_i \cup \{d\}$ so that $<_1 \cap <_2 = <_P$ if $<_P$ has the Hasse diagram as pictured in Figure 5 (C), namely

$$(II') \quad c_{5i} <_1 d <_1 c_{5i+1} <_1 c_{5i+2} <_1 c_{5i+4} <_1 c_{5i+3} \quad \text{and} \\ c_{5i+2} <_2 c_{5i+3} <_2 d <_2 c_{5i+4} <_2 c_{5i} <_2 c_{5i+1}.$$

Now to complete our definition of $<_P$ on ω , we proceed in stages as follows.

STAGE 0:

If $i \in A^0$, let $C_{i-1} <_P \{d_0\} <_P C_{i+1}$, and define $<_P$ on $C_i \cup \{d_0\}$ so that we have a Hasse diagram as in Figure 5 (B). If $i \in B^0$, let $C_{i-1} <_P \{d_0\} <_P C_{i+1}$ and define $<_P$ on $C_i \cup \{d_0\}$ so that we have a Hasse diagram as in Figure 5 (C). If $A^0 \cup B^0 = \emptyset$, define $\{d_0\} <_P C$. Note this defines $<_P$ on all of $C \cup \{d_0\}$ by transitivity.

STAGE $s > 0$:

Assume we have defined $<_P$ on $C \cup \{d_0, \dots, d_{s-1}\}$ so that for all $j < s$, $C_{i-1} <_P \{d_j\} <_P C_{i+1}$ if $i \in (A^j \cup B^j) \setminus (A^{j-1} \cup B^{j-1})$ and $\{d_j\} <_P C \cup \{d_0, \dots, d_{j-1}\}$ otherwise. Then if $i \in A^s \setminus A^{s-1}$, let $C_{i-1} <_P \{d_s\} <_P C_{i+1}$ and define $<_P$ on $C_i \cup \{d_s\}$ so that we have a Hasse diagram as pictured in

Figure 5 (B). If $i \in B^s \setminus B^{s-1}$, let $C_{i-1} <_P \{d_s\} <_P C_{i+1}$ and define $<_P$ on $C_i \cup \{b_s\}$ so that we have a Hasse diagram as pictured in Figure 5 (C). If $(A^s \cup B^s) \setminus (A^{s-1} \cup B^{s-1}) = \emptyset$, define $\{d_s\} <_P C \cup \{d_0, \dots, d_{s-1}\}$. Again this defines $<_P$ on all of $C \cup \{d_0, \dots, d_s\}$ by transitivity.

This completes our definition of $<_P$ on ω . It is easy to see that the definition of $<_P$ is completely effective. Given our remarks prior to our definition the stages, it is routine to check that up to a permutation of indices, if $<_1$ and $<_2$ are two linear orderings of ω , then $<_1 \cap <_2 = <_P$ if and only if $A \subseteq S(<_1, <_2)$ and $B \cap S(<_1, <_2) = \emptyset$.

This completes the proof of Theorem 7.3. $\square$

As in Section 6, we have a number of immediate corollaries and we state only a few.

Theorem 7.4

- (a) *There is a recursive poset of width k which has no covering by k chains.*
- (b) *There is a recursive poset $\mathcal{A}$ of height k such that any two distinct coverings of $\mathcal{A}$ by k antichains are Turing incomparable, where distinct means not obtainable from the other by a permutation of the antichains in combination with the shifting of a finite number of elements.*
- (c) *If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} \leq_T \mathbf{0}'$, then there is a recursive poset $\mathcal{A} = (A, R)$ of dimension d , but not of recursive dimension d such that there exists a set $\{(A, L_1), \dots, (A, L_d)\}$ of degree $\mathbf{a}$ of linear orderings such that $R = L_1 \cap \dots \cap L_d$.*

Despite the fact that we were only able to prove that problems considered in this section can represent an arbitrary classes of separating sets for a pair of r.e. sets, we can still use this representation to prove results on index sets by transferring our results on index sets for separating sets given in Theorem 2.49. We will use the example of covering a poset with k -chains. It is easy to see that we can effectively list all primitive recursive posets with universe ω as $PO_0, PO_1, \dots$. There is a recursive function h such that for any given poset PO_e , there is a one-to-one degree preserving correspondence between the set of coverings of PO_e with k -chains and the set of infinite paths through $T_{h(e)}$. Similarly given any pair of r.e. sets W_a and W_b , we can use the construction of Theorem 7.3 to show that there exists a recursive function g

such that there is an effective one-to-one degree preserving correspondence between the set of separating sets $SS(a, b)$ and the set of coverings of $PO_{g(e)}$ with k -chains. Actually one must modify the construction of Theorem 7.3 slightly to ensure that the poset produced is primitive recursive but this is just a matter of padding the construction. Examples of this type of padding are given in Section 12. We can then use these recursive functions h and g to translate the results of Theorem 2.49 to index set results for coverings with k -chains. The idea is to use h to prove the bounds on the complexity and to use g to establish the completeness.

Theorem 7.5

- (a) $(\{e : PO_e \text{ cannot be covered by } k\text{-chains}\},$
 $\{e : PO_e \text{ can be covered by } k\text{-chains}\})$
 is (Σ_1^0, Π_1^0) complete.
- (b) For any positive integer c ,
 $(\{e : PO_e \text{ has } > c \text{ coverings by } k\text{-chains}\},$
 $\{e : PO_e \text{ has } \leq c \text{ coverings by } k\text{-chains}\})$
 is (Σ_2^0, Π_2^0) complete,
 $\{e : PO_e \text{ has exactly } c = 1 \text{ coverings by } k\text{-chains}\}$ is D_2^0 complete,
 and
 $\{e : PO_e \text{ has exactly 1 covering by } k\text{-chains}\}$ is Π_2^0 complete.
- (c) $(\{e : PO_e \text{ has only finitely many coverings by } k\text{-chains}\},$
 $\{e : PO_e \text{ has infinitely many coverings by } k\text{-chains}\})$
 is (Σ_3^0, Π_3^0) complete.
- (d) $(\{e : PO_e \text{ has a recursive covering by } k\text{-chains}\},$
 $\{e : PO_e \text{ has no recursive covering by } k\text{-chains}\})$
 is (Σ_3^0, Π_3^0) complete, and
 $\{e : PO_e \text{ has a covering by } k\text{-chains},$
 $\text{but has no recursive covering by } k\text{-chains}\}$
 is Π_3^0 complete.

Similar results hold for the other problems discussed in this section.

8 Recursive linear orderings and ordered structures

There are several problems associated with linear orderings which lead to Π_1^0 classes. Here we encounter Π_1^0 classes which are not necessarily recursively bounded. There are two types of problems considered here. The problem with an ordered structure is to find a linear ordering consistent with the structure, for example, an ordering which will turn a group into an ordered group. The second type of problem is concerned with pure orderings. Here we are looking for sub-orderings of type ω , for ω -successivities, and for self-embeddings.

8.1 Recursive ordered structures

In this subsection, we consider two problems:

- (1) The problem of finding an ordering of an Abelian group.
- (2) The problem of finding an ordering of a formally real field.

In each case, the set of solutions to a given effective problem can always be represented by an r.b. Π_1^0 class and in case (2), any r.b. Π_1^0 class can be represented by such a set.

In this section, we will assume that a recursively presented group, ring, or field is given by recursive addition, subtraction, multiplication and division functions on the set ω , as appropriate. An *r.e. ring* is the quotient of a recursive ring modulo an r.e. ideal, and an *r.e. group* is the quotient of a recursive group modulo an r.e. normal subgroup. An ordering will be represented by the cone of positive elements.

A formally real field is a field F such that no sum of (non-zero) squares equals zero. A field $(F, +^F, \cdot^F)$ is said to be *ordered* by the relation $\leq$ provided that $\leq$ is a linear ordering such that for all $a, b, c \in F$,

$$(i) \quad a \leq b \rightarrow a +^F c \leq b +^F c.$$

$$(ii) \quad (0 \leq a \ \& \ 0 \leq b) \rightarrow 0 \leq a \cdot^F b.$$

An ordering for a commutative group $(G, +^G, 0^G)$ is defined similarly except in this case the ordering $\leq$ need only satisfy condition (i).

The set $C = C_{\leq} = \{a \in F : 0 \leq a\}$ clearly satisfies the following for any $a, b \in F$:

- (i) $a, b \in C \rightarrow a +^F b \in C$.
- (ii) $a, b \in C \rightarrow a \cdot^F b \in C$.
- (iii) $(a \in C \ \& \ 0^F -^F a \in C) \iff a = 0^F$.
- (iv) $a \in C \vee 0^F -^F a \in C$.

A subset C of F satisfying (i) to (iv) is said to be a *positive cone* of F . Thus any linear ordering of F defines a positive cone and conversely any positive cone C of F defines a linear ordering by

$$a \leq b \iff b -^F a \in C.$$

Thus we will identify the set of linear orderings of a field F with the set of positive cones of F .

For a commutative group $(G, +^G, 0^G)$, a cone C need only satisfy (i), (iii) and (iv).

The classical result of Artin-Schreier [6] is that any formally real field can be ordered. Craven showed in [32] that any closed subset C of the Cantor space can be represented as the set of orderings of some formally real field F . Metakides and Nerode [110] made this proof effective by showing that if C is a Π_1^0 class, then F may be taken to be a recursive field. Downey and Kurtz observed that the field F may have additional orderings which are compatible with the group structure although not compatible with the field structure. Thus the similar problem for orderings of groups remains open. Downey and Kurtz constructed in [42] a recursive group isomorphic to $\oplus_{\omega} \mathbb{Z}$ which has no recursive ordering. The classical result here of Levi [96] is that an Abelian group can be ordered if and only if it is torsion-free.

Theorem 8.1 *For each specific r.e. instance of the problems (1) and (2) listed above, the set of solutions can be represented as an r.b. Π_1^0 class.*

Proof. For recursive structures, this is immediate from the discussion above. For an r.e. structure, say $F = R/I$, observe that a positive cone C on R/I corresponds to a subset C' of R satisfying clauses (i), (ii) and (iv) along with the following modified version of clause (iii).

(iii) $(a \in C' \ \& \ 0^F -^F a \in C') \iff a \in I.$

We leave it to the reader to translate these four clauses into a definition of a recursive tree T such that $[T]$ represents the set of positive cones on F . The proof for ordered groups is similar. $\square$

We can as usual derive a number of immediate corollaries from the results of Section 2. For example,

Theorem 8.2

- (a) *Any r.e. presented group which has an ordering has an ordering of r.e. degree.*
- (b) *If the set of orderings of the r.e. presented group G is countably infinite and nonempty, then G has a recursive ordering.*
- (c) *If the r.e. presented field F has only finitely many orderings, then every ordering of F is recursive.*

Next we turn to the other direction of our correspondence, that is, representing an arbitrary Π_1^0 class by the set of solutions to certain of these problems. The problem of orderings of formally real fields was solved by Metakides and Nerode in [112].

Theorem 8.3 *Any r.b. Π_1^0 class P can be represented by the set of orderings of a formally real field.*

Proof. Let the recursive tree $T \subseteq {}^{<\omega}0, 1$ be given so that $P = [T]$. The construction begins with the underlying ring $R = \mathbb{Q}[x_i : i \in \omega]$ (the ring of polynomials with rational coefficients in infinitely many variables). We define a recursive maximal ideal of R such that the set of orderings of the field R/I represents $[T]$. We sketch a proof in which is somewhat different from that in [112].

The first step of our construction is to adjoin to $\mathbb{Q}$ the radicals $\sqrt{p_i}$, where p_i is the i -th prime. That is, we put $x_i^2 - p_i$ into I for each i . Thus we have initially a continuum of possible orderings on $\mathbb{Q}[\sqrt{p_i} : i < \omega]$, where to each $\Pi \in \{0, 1\}^\omega$ there corresponds the ordering $R(\Pi)$ determined by taking $x_i > 0$ if $\Pi(i) = 0$ and $x_i < 0$ if $\Pi(i) = 1$. Now for any $\sigma \notin T$, we use an auxiliary variable y_σ to eliminate the ordering corresponding to σ in

the following manner. We uniformly and effectively define, for σ of length n , a polynomial $f_\sigma(x_0, \dots, x_{n-1})$ such that for $(e_0, \dots, e_{n-1}) \in \{0, 1\}^n$,

$$f_\sigma((-1)^{e_0}\sqrt{2}, (-1)^{e_1}\sqrt{3}, \dots, (-1)^{e_{n-1}}\sqrt{p_{n-1}}) < 0$$

if and only if $(e_0, \dots, e_{n-1}) = \sigma$. Then we add to I the polynomial $y_\sigma^2 = f_\sigma(x_0, \dots, x_{n-1})$ thus adjoining to our field a square root for $f_\sigma(x_0, \dots, x_{n-1})$. It follows that if $\sigma \prec \Pi$, then the ordering $R(\Pi)$ is not compatible with the field, since it forces a negative element to have a square root. The function f_σ is defined to be

$$f_\sigma(x_0, \dots, x_{n-1}) = c_\sigma - (-1)^{\sigma(0)}x_0 - \dots - (-1)^{\sigma(n-1)}x_{n-1},$$

where c_σ is the least integer c such that $\sqrt{2} + \sqrt{3} \cdots + \sqrt{p_{n-1}} > c$.

(For example, suppose $\sigma = (0, 1)$. Then we want $f_\sigma(\sqrt{2}, -\sqrt{3}) < 0$, $f_\sigma(\sqrt{2}, \sqrt{3}) > 0$, $f_\sigma(-\sqrt{2}, -\sqrt{3}) > 0$, and $f_\sigma(-\sqrt{2}, \sqrt{3}) > 0$. We compute that $3 < \sqrt{2} + \sqrt{3} < 4$ and define $f_\sigma(x_0, x_1) = 3 - x_0 + x_1$.)

Finally, to prevent any additional orderings from arising due to the new roots in the field, we add a sequence of roots $y_{i,j}$ to the field such that $y_{i,0} = y_i$ and $y_{i,j+1}^2 = y_{i,j}$. Thus each y_i and each $y_{i,j}$ is forced to be positive. $\square$

This representation theorem has, as usual, a number of immediate corollaries of which we state only a few.

Theorem 8.4

- (a) *There is a recursive formally real field which has no recursive ordering.*
- (b) *There is a recursive formally real field which has continuum many orderings and such that any two distinct orderings are Turing incomparable.*
- (c) *There is a recursive formally real field F such that if $\mathbf{a}$ is the degree of any ordering of F and $\mathbf{b}$ is an r.e. degree with $\mathbf{a} \leq_T \mathbf{b}$, then $\mathbf{b} =_T \mathbf{0}'$.*
- (d) *There is a recursive formally real field F which has a unique non-recursive ordering $\leq_0$, such that this ordering $\leq_0$ has degree $\mathbf{0}'$, and such that for any other ordering $\leq$ of F , there is some finite subset A of F such that for any ordering $\leq'$ of R , if $\leq$ agrees with $\leq'$ on A , then $\leq = \leq'$.*

Recently, D. Reed Solomon [144] showed that the analogue of the Metakides-Nerode theorem fails for Abelian groups, that is, every abelian group has either two or has infinitely many orderings.

8.2 Recursive Linear Orderings

There are three problems discussed in this subsection related to a given recursive linear ordering $\mathcal{A} = (A, \leq^A)$.

- (1) The problem of finding a subordering of $\mathcal{A}$ of type ω or of type ω^* .
- (2) The problem of finding an ω -successivity or an ω^* -successivity in $\mathcal{A}$.
- (3) The problem of finding a self-embedding of $\mathcal{A}$.

(1) Suborderings of type ω or ω^* .

A standard classical result is that any infinite linear ordering has a subordering $\{f(0), f(1), \dots\}$ of order type either ω or ω^* (the order type of the negative integers). Tennenbaum and independently Denisov showed that there is an infinite recursive linear ordering of order type $\omega + \omega^*$ which has no recursively enumerable subordering of either type (see Rosenstein [132] or Downey [39]). The suborderings of type ω (respectively ω^*) are simply the functions $f : \omega \rightarrow A$ such that $f(n) \leq^A f(n+1)$ (resp. $f(n+1) \leq^A f(n)$) for all n . Thus in each case the set of solutions to the problem of finding such a subordering is a Π_1^0 class, but is clearly not bounded. For example, if $\mathcal{A}$ is the standard ordering $(\omega, \leq)$, then the class of suborderings of $\mathcal{A}$ of type ω is just the class of all increasing sequences of natural numbers, which is homeomorphic to ω^ω and not even compact. We observe that the class of suborderings of type ω is always a perfect set, since for any such subordering f and any n , there is another subordering of type ω given by $(f(0), f(1), \dots, f(n), f(n+2), f(n+4), \dots)$.

Theorem 8.5 *For any recursive linear ordering $\mathcal{A} = (A, \leq^A)$, the class of suborderings of $\mathcal{A}$ of type ω (respectively, of type ω^*) is a perfect Π_1^0 class.*

Thus all we can say is that if a recursive linear ordering has a subordering of type ω (respectively, type ω^*), then it has such a subordering which is recursive in some Σ_1^1 set. It was shown by Manaster that any recursive linear ordering has a Π_1^0 subordering of type ω or of type ω^* (see Downey [39]).

(2) Successivities.

An element b of $\mathcal{A}$ is said to be the *successor* of an element a if $a <^A b$ and there is no c such that $a <^A c <^A b$; in such a case, we write $b = S_A(a)$. We say that a subordering f of type ω in $\mathcal{A}$ is an ω -*successivity* if $f(n+1)$ is the successor of $f(n)$ in the linear ordering for each n , and similarly define an ω^* -*successivity*. Then the family P of ω -successivities is a Π_1^0 class and likewise the family of ω^* -successivities.

Observe that the class of ω -successivities of the standard ordering on ω consists of all sequences $(n, n+1, n+2, \dots)$, and is thus a countable set in which all elements are isolated. As for the suborderings above, this class is not necessarily compact.

In general, there is at most one ω -successivity f for each starting element $f(0) = a$, so that every member of the class P of ω -successivities is isolated; a class with this property is said to be *scattered*. Clearly P is also countable. Furthermore, we can define a bounded recursive tree T with $P = [T]$ by $(a_0, a_1, \dots, a_n) \in T$ if and only if

$$(\forall i < n) [a_i <_A a_{i+1} \ \& \ (\forall m < a_{i+1}) \neg(a_i <_A m <_A a_{i+1})].$$

A similar argument applies for ω^* -successivities.

Theorem 8.6 *For any recursive linear ordering $\mathcal{A} = (A, \leq^A)$, the class of ω -successivities (respectively ω^* -successivities) of $\mathcal{A}$ is a scattered, bounded Π_1^0 class.*

Applying Theorem 2.12, we have the following.

Corollary 8.7 *Every ω -successivity (respectively ω^* -successivity) of a recursive linear ordering $\mathcal{A}$ is recursive in $\mathbf{0}'$.*

This of course may also be proven directly from the definition of a successivity. It follows from the result of Tennenbaum and Denisov that there is a recursive linear ordering of type $\omega + \omega^*$ which has no recursive ω -successivity.

(3) Self-embeddings.

Another classical result is due to Dushnik and Miller [46], who showed that an infinite countable linear ordering always has a non-trivial self-embedding. Hay, Manaster and Rosenstein [65] constructed a recursive linear ordering of

type ω with no non-trivial recursive self-embedding. A map $f : A \rightarrow A$ is a self-embedding of $\mathcal{A}$ if, for all a and b , $f(a) \leq^A f(b)$ if and only if $a \leq b$. The family of self-embeddings of a recursive linear ordering is again seen to be a Π_1^0 class. For the standard ordering on ω , it is clear that a self-embedding is the same thing as a subordering of type ω . Thus the class of self-embeddings need not be compact.

Now $\mathcal{A}$ always has a recursive self-embedding, namely the identity function. If $\mathcal{A}$ has a non-trivial self-embedding, then we can fix an element a and consider the Π_1^0 class of self-embeddings f such that $f(a) \neq a$. It follows as usual that $\mathcal{A}$ at least has a non-trivial self-embedding which is recursive in some Σ_1^1 set.

Theorem 8.8 *For any recursive linear ordering $\mathcal{A} = (A, \leq^A)$, the class of self-embeddings of $\mathcal{A}$ is a Π_1^0 class.*

Theorem 8.9 *For any recursive linear ordering $\mathcal{A} = (A, \leq^A)$, if $\mathcal{A}$ has a non-trivial self-embedding, then $\mathcal{A}$ has a self-embedding recursive in a Σ_1^1 set.*

Recently, Downey and Lempp [43] showed that the proof-theoretical strength of the Dushnik-Miller theorem is ACA, which implies that every recursive linear ordering has a self-embedding which is recursive in $\mathbf{0}'$.

9 Recursive Analysis

In this section we sketch the relation between Π_1^0 classes and real numbers and functions. The problems we consider include finding zeroes, extreme values and fixed points of recursively continuous functions. We also consider Julia sets and basins of attraction for recursively continuous dynamical systems. For details and further results, see Ko [88] or [27]. More specifically, the problems that we will consider are the following.

(1) Zeroes of continuous functions.

The classical problem here is to find a zero for a continuous function. The intermediate value theorem can be used to show the existence of a zero for a continuous function which is negative at one point and positive at another point. The effective version of this theorem also holds, that is, any recursive function on the reals which is negative at one point and positive at another

point has a recursive zero, which can be computed by repeatedly splitting the interval between the two initial points. (See Pour-El-Richards [123] for a proof.) However, Lacombe [94] showed that there are recursive functions which have zeroes but have no recursive zeroes. We will give the improvement of this result due to Nerode and Huang [115] by showing that every Π_1^0 class is the set of zeroes of some recursive function.

(2) The Extreme Value Theorem.

The classical result here is that any function which is continuous on a compact set takes on a maximum and a minimum on that set. The problem here is to find a point where the maximum or minimum is attained. Lacombe [94] showed that the extreme values of a recursive function on $[0, 1]$ are themselves recursive and also constructed a recursive function F on $[0, 1]$ which does not attain its maximum at any recursive point. We will present the result of Nerode and Huang [115] that any Π_1^0 class may be represented as the set of points where some recursively continuous function attains its maximum.

(3) Fixed points of continuous functions.

The problem here is to find a fixed point for a given continuous function. A simple application of the intermediate value theorem shows that any continuous function F on $[0, 1]$ has a fixed point. It is well known that if F is recursively continuous, then F will have a recursive fixed point. The Brouwer Fixed Point Theorem says that a continuous function on $[0, 1] \times [0, 1]$ will also have a fixed point, but Orevkov [122] showed that there need not be a recursive fixed point. Results for other spaces are different. On the real line, the continuous function $F(x) = x + 1$ has no fixed point. On ω^ω , the function $F((x(0), x(1), \dots)) = (1 + x(0), 1 + x(1), \dots)$ has no fixed point. On the Cantor space the function $F(x(0), x(1), \dots) = (1 - x(0), 1 - x(1), \dots)$ has no fixed point.

(4) Dynamical systems.

We will give a few results on effective real dynamical systems from Cenzer [14] and from Ko [89]. We shall view a dynamical system as determined by a continuous function F on a space X . The associated problem is to determine the behavior of the sequence $x, F(x), F(F(x)), \dots$ for a given x . In particular, we want to find those points x for which this sequence is bounded

or converges to some finite number and those x for which the sequence is unbounded or diverges to infinity where X is either the real line or the Baire space. If F is a polynomial, then it is always possible to compute a bound c such that $\{F^{(n)}(x) : n < \omega\}$ is bounded if and only if $|F^{(n)}(x)| < c$ for all n . In fact, we can take c large enough so that $F(x) > x + 1$ for all $x > c$, so that $\lim_{n \rightarrow \infty} F^{(n)}(x) = \infty$ for all $x > c$. In this situation, we say that ∞ is an *attracting point* for F . Then $\{x : |F^{(n)}(x)| \leq c \text{ for all } n\}$ is called the *Julia set* of F . (See Blum, Shub and Smale [11].) It is then easy to see that the Julia set of any continuous function must be a compact set and we will show that for a recursively continuous function, the Julia set is a Π_1^0 class. The first problem for dynamical systems is to find a member of the Julia set.

A point x is said to be a *periodic point* of a continuous function F if $F^{(n)}(x) = x$ for some finite n . The *basin of attraction* $B(x)$ of x is defined to be $\{u : \lim_n F^{(n)}(u) = x\}$. The periodic point x is said to be *attracting* if there is some open neighborhood U about x such that $U \subseteq B(x)$. The basin of attraction of infinity may also be defined as $\{u : \lim_n F^{(n)}(u) = \infty\}$. Thus the basin of attraction is an open set. We will show that for a recursively continuous function, the complement of a basin of attraction is a Π_1^0 class. If 1 is an attracting periodic point of a function F on $\{0, 1\}^\omega$ or $[0, 1]$, then we will refer to the complement of $B(1)$ as the Julia set of F . The problem here is to find a point not in the basin of attraction.

Before turning to the problems mentioned above, we give a brief introduction to recursive analysis, including the problem of characterizing the recursive image of the interval and the related concept of a real as a Dedekind cut of rationals, which was studied by Soare in [142, 141].

A basic principle of recursive analysis is that a recursive function on the real numbers is an effectively continuous function and a Π_1^0 class is an effectively closed set. We will consider the real line $\mathbb{R}$, as well as three subspaces: the space of irrationals, which is homeomorphic to the Baire space ω^ω and two compact subspaces, the interval $[0, 1]$ and the Cantor space, which is recursively homeomorphic to $\{0, 1\}^\omega$. Since $\mathbb{R}$ is recursively homeomorphic to the open interval $(0, 1)$ via the order-preserving map $\frac{e^x}{1+e^x}$, we will frequently identify $\mathbb{R}$ with $(0, 1)$ and treat it as a subset of $[0, 1]$.

Let $\mathcal{D}$ be the set of dyadic rationals in $[0, 1]$. Then $[0, 1]$ has a basis of open intervals (a, b) , $[0, c)$ or $(d, 1]$ where $a, b, c, d \in \mathcal{D}$. Thus an open subset of $[0, 1]$ is a countable union

$$U = \bigcup_n (a_n, b_n) \bigcup_n [0, c_n) \bigcup_n (d_n, 1]$$

of dyadic intervals. The open set U is said to be *effectively open*, or Σ_1^0 , if the sequences a_n, b_n, c_n and d_n are recursive. Then a closed set C is said to be *effectively closed*, or Π_1^0 , if it is the complement of an effectively open set.

Any $x \in \{0, 1\}^\omega$ represents a real $r_x = \sum_n x(n)/2^n \in [0, 1]$. In addition, for any $\sigma \in \{0, 1\}^{<\omega}$, $\sigma \frown 0^\omega$ represents the dyadic rational $q_\sigma = \sum_{i < n} \sigma(i)/2^i$. Some difficulty arises from the fact that q_σ has another representation, $\sigma \frown (n-1) \frown 0 \frown 1^\omega$ (assuming that σ ends in a 1). Each dyadic rational is of course recursive, so that we may unambiguously say that r is a *recursive real* if $r = r_x$ for some recursive sequence $x \in \{0, 1\}^\omega$. Then a subset P of $\{0, 1\}^\omega$ represents a subset of $[0, 1]$ if and only if, for all x, y such that $r_x = r_y$, we have $x \in P$ if and only if $y \in P$. For any $\sigma \in \{0, 1\}^{<\omega}$ of length n , the members of $I(\sigma)$ represent the members of the real closed interval $[q_\sigma, q_\sigma + 2^{-n}]$, which we denote by $U(\sigma)$. More generally, if $r < s$ are recursive reals, then the interval $[r, s]$ is a Π_1^0 class, since, if $r = r_x$ and $s = r_y$, then

$$r_z \in [r, s] \iff (\forall n) [q_{x \upharpoonright n} - 2^{-n} \leq q_{z \upharpoonright n} \leq q_{y \upharpoonright n} + 2^{-n}].$$

Lemma 9.1

(a) *The following are equivalent for any subset K of $[0, 1]$.*

- (1) *K is a Π_1^0 class.*
- (2) *K is closed and $\{\langle p, r \rangle \in \mathcal{D}^2 : K \cap [p, r] = \emptyset\}$ is an r.e. set.*
- (3) *K is represented by a Π_1^0 class $P \subset \{0, 1\}^\omega$.*

(b) *K may be represented by a recursive binary tree with no dead ends if and only if $\{\langle p, r \rangle \in \mathcal{D}^2 : K \cap [p, r] = \emptyset\}$ is recursive.*

Proof.

(a) We show that both (1) and (3) are equivalent to (2). Suppose first that K is a Π_1^0 class and let

$$[0, 1] \setminus K = \bigcup_n (a_n, b_n) \cup \bigcup_n [0, c_n) \cup \bigcup_n (d_n, 1].$$

Then

$$K \cap [p, r] = \emptyset \iff (\exists n) [p, r] \subseteq \bigcup_{m < n} (a_m, b_m) \cup \bigcup_{m < n} [0, c_m) \cup \bigcup_{m < n} (d_m, 1].$$

Suppose next that $A = \{\langle p, r \rangle : K \cap [p, r] = \emptyset\}$ is an r.e. set. Then K is a Π_1^0 class since

$$[0, 1] \setminus K = \bigcup \{\langle p, r \rangle : \langle p, r \rangle \in A\}.$$

Furthermore, K is represented by $[T]$ where the Π_1^0 tree T is defined as follows. Given σ of length n , let

$$\sigma \in T \iff [q_\sigma, q_\sigma + 2^{-n}] \not\subseteq \bigcup \{\langle p, r \rangle : \langle p, r \rangle \in A^n\}.$$

(Here we replace $q + 2^{-n}$ with 1 if $q = 1$.)

Finally suppose that $K = \{r_x : x \in P\}$ for some Π_1^0 class $P = [T] \subseteq \{0, 1\}^\omega$. Then for any σ ,

$$K \cap [q_\sigma, q_\sigma + 2^{-n}] = \emptyset \iff \sigma \notin \text{Ext}(T)$$

Since any dyadic interval $[p, r]$ may be decomposed into a finite union of intervals of the form $[q_\sigma, q_\sigma + 2^{-n}]$, it follows that $\{\langle p, r \rangle : K \cap [p, r] = \emptyset\}$ is an r.e. set.

(b) This follows from the observation that, if K is represented by $[T]$, then $\sigma \in \text{Ext}(T) \iff K \cap [q_\sigma, q_\sigma + 2^{-|\sigma|}] \neq \emptyset$. $\square$

An arbitrary Π_1^0 class $P \subseteq \{0, 1\}^\omega$ can be represented by a Π_1^0 subclass of $[0, 1]$ by the following lemma.

Lemma 9.2 *For any Π_1^0 class $P \subseteq \{0, 1\}^\omega$, there is a Π_1^0 subclass $Q \subseteq \{0, 1\}^\omega$ which represents a subset of $[0, 1] \setminus \mathcal{D}$ which is recursively homeomorphic to P .*

Proof. Let the recursive homeomorphism φ be defined by

$$\varphi(x(0), x(1), \dots) = (1, 0, x(0), 1, 0, x(1), \dots)$$

and let $Q = \varphi[P]$. Q represents a subset of $[0, 1]$ since every element of Q has both infinitely many “1’s” and infinitely many “0’s”. $\square$

We can characterize those intervals which are Π_1^0 classes using the notion of the *Dedekind cut* $L(r) = \{q \in \mathcal{D} : q \leq r\}$ of a real number r . Soare showed in [142, 141] that if $x \in \{0, 1\}^\omega$ is the characteristic function of a Π_1^0 set (respectively a Σ_1^0 set), then $L(r_x)$ is a Π_1^0 set (resp. a Σ_1^0 set) and that these implications are not reversible.

The set $\omega^{<\omega}$ and the space ω^ω may be linearly ordered by the lexicographic ordering $\leq_L$, where $x <_L y$ if, for some n , $x(n) < y(n)$ and $x(i) = y(i)$ for all $i < n$. This ordering is recursive on $\omega^{<\omega}$ and thus is Π_1^0 on ω^ω , since $x \leq_L y \iff (\forall n)x \upharpoonright n \leq y \upharpoonright n$.

We now define the interval $[x, y] = \{z : x \leq_L z \leq_L y\}$ and also $[x, \infty] = \{z : x \leq_L z\}$. Then we let $L(x) = \{\sigma \in \omega^{<\omega} : \sigma \frown 0^\omega \leq x\}$. These notions may also be restricted to $\{0, 1\}^\omega$ and $\{0, 1\}^{<\omega}$. Observe that for non-dyadic rationals r_x and r_y , $r_x < r_y$ if and only if $x <_L y$.

Lemma 9.3

- (a) For any $x < y$ in either $[0, 1]$, $\{0, 1\}^\omega$, or ω^ω , the interval $[x, y]$ is a Π_1^0 class if and only if $L(x)$ is a Σ_1^0 set and $L(y)$ is a Π_1^0 set.
- (b) In either $[0, 1]$, $\{0, 1\}^\omega$, or ω^ω , $L(x)$ is a recursive set (respectively recursive in A) if and only if x is recursive (resp. in A).
- (c) For any $x \in \{0, 1\}^\omega$, if x is the characteristic function of a $\Sigma_1^{0,A}$ (respectively $\Pi_1^{0,A}$) set, then $L(x)$ is a $\Sigma_1^{0,A}$ (resp. $\Pi_1^{0,A}$) set.

Proof.

(a) First consider the case, where $x < y$ and $x, y \in \omega^\omega$. We claim that $[x, y]$ is a Π_1^0 class if and only if both $[x, \infty]$ and $[0, y]$ are Π_1^0 classes. The if direction follows from the fact that $[x, y] = [x, \infty] \cap [0, y]$. For the other direction, choose $\sigma \in \omega^\omega$ and n such that $x \upharpoonright n \leq_L \sigma <_L y \upharpoonright n$ and observe that $[x, \infty] = [x, y] \cup [\sigma \frown 1^\omega, \infty]$ and $[0, y] = [0, \sigma \frown 0^\omega] \cup [x, y]$.

Thus we need only show that $[x, \infty]$ is a Π_1^0 class iff $L(x)$ is a Σ_1^0 set and that $[0, y]$ is a Π_1^0 class iff $L(x)$ is a Π_1^0 set. Suppose that $[x, \infty] = [T]$ for some recursive tree T . Then

$$\sigma \in L(x) \iff \sigma \frown 0^\omega \notin [T] \iff (\exists n) \sigma \frown 0^n \notin T$$

and hence $L(x)$ is Σ_1^0 set. Vice versa, suppose that $L(x)$ is a Σ_1^0 set. Then we have

$$z \in [x, \infty] \iff (\forall m)(z \upharpoonright m \notin L(x))$$

so that $[x, \infty]$ is a Π_1^0 class.

Similarly, if $[0, y] = [T]$ for some recursive tree, then

$$\sigma \in L(y) \iff (\forall n)(\sigma \frown 0^n \in T)$$

so that $L(y)$ is a Π_1^0 set. Vice versa, if $L(y)$ is a Π_1^0 set, then

$$z \in [0, y] \iff (\forall n)(z \upharpoonright n \in L(y))$$

so that $[0, y]$ is a Π_1^0 class.

For $x, y \in \{0, 1\}^\omega$, the argument is similar, except that $[x, \infty]$ is replaced by $[x, 1^\omega]$.

For $r_x, r_y \in [0, 1]$, the problem reduces to the previous case of $\{0, 1\}^\omega$, as long as we take x to end in 0^ω whenever $r_x \in \mathcal{D}$ and y to end in 1^ω whenever $r_y \in \mathcal{D}$, so that $q_\sigma \in [r_x, r_y] \iff \sigma \in [x, y]$.

(b) We give the argument for ω^ω . $L(x)$ is recursive in x , since

$$\sigma \in L(x) \iff \sigma \leq_L x \upharpoonright |\sigma|.$$

Also, x is recursive in $L(x)$, since for each n , $x(n+1)$ is the least a such that $x \upharpoonright n \smallfrown a \in L(x)$ & $x \upharpoonright n \smallfrown a + 1 \notin L(x)$.

(c) Now suppose that x is the characteristic function of a Π_1^0 set, i.e., x is the characteristic function of $\omega \setminus A$ where A is an r.e. set. Then let A^s for $s \geq 0$ be some effective enumeration of A . Thus x is the decreasing limit of a sequence $(x_0, x_1, \dots)$ where x_s is the characteristic function of A^s . Then

$$\sigma \in L(x) \iff (\forall n)(\sigma \leq_L x_n \upharpoonright |\sigma|).$$

Similarly, if x is the characteristic function of a Σ_1^0 set A then x is the increasing limit of the sequence (x_n) . Hence $\sigma \in L(x) \iff (\exists n)(\sigma \leq_L x_n \upharpoonright |\sigma|)$. $\square$

It follows from part (b) that $L(x)$ is Δ_2^0 if and only if x is Δ_2^0 , and that if x is Π_2^0 (respectively, Σ_2^0), then $L(x)$ is Π_2^0 (resp. Σ_2^0).

Theorem 9.4

- (a) Let $x \in \omega^\omega$. If x is the maximum element of an r.b. Π_1^0 class, $L(x)$ is a Π_1^0 set. If $L(x)$ is a Π_1^0 set and, in addition, x is not hyperimmune, i.e., there is a recursive function f such that $x(e) \leq f(e)$ for all e , then x is the maximum element of some r.b. Π_1^0 class. If x is the minimum element of some r.b. Π_1^0 class, then $L(x)$ is a Σ_1^0 set. If $L(x)$ is a Σ_1^0 set and, in addition, x is not hyperimmune, then x is the minimum element of some r.b. Π_1^0 class.

- (b) For any x in $[0, 1]$, x is the maximum element of some Π_1^0 class if and only if $L(x)$ is Π_1^0 , and x is the minimum element of some Π_1^0 class if and only if $L(x)$ is Σ_1^0 .
- (c) For any $x \in \omega^\omega$ or $[0, 1]$, x is the maximum (respectively, minimum) element of a Π_1^0 class represented by a tree with no dead ends if and only if x is recursive.
- (d) For any $x \in \omega^\omega$, if x is the maximum element of a bounded Π_1^0 class, then $L(x)$ is a Π_2^0 set and if x is the minimum element of a bounded Π_1^0 class, then $L(x)$ is a Σ_2^0 set.
- (e) For any $x \in \omega^\omega$, if x is the maximum element of a Π_1^0 class, then $L(x)$ is a Σ_1^1 set, and if x is the minimum element of a Γ_1^0 class, then $L(x)$ is a Π_1^1 set.

Proof. We just give proofs for the maximum element versions.

(a) Suppose that $L(x)$ is a Π_1^0 set and there is a recursive function f such that $x(e) \leq f(e)$ for all e . Then x is the maximum element of the Π_1^0 interval $[0, x]$ by Lemma 9.3. Hence x is the maximal element of the r.b. Π_1^0 class $[0, x] \cap [T]$ where T is the recursive tree such that

$$\sigma \in T \iff (\forall i \leq |\sigma|) (\sigma(i) \leq f(i)).$$

Now let x be the maximum element of an r.b. Π_1^0 class $P = [T]$. Then $\sigma \in L(x)$ if and only if

$$(\exists y) [y \in P \ \& \ \sigma \leq_L y] \iff (\exists \tau \in \omega^{|\sigma|}) [\tau \in \text{Ext}(T) \ \& \ \sigma \leq_L \tau].$$

Since T is r.b., the search for τ is bounded and, since $\text{Ext}(T)$ is a Π_1^0 set, $L(x)$ is a Π_1^0 set.

If T has no dead ends, then $\text{Ext}(T)$ is recursive, so that $L(x)$ is recursive. This completes the proof of part (a) as well as part (c).

Part (b) now follows from Lemma 9.1.

Parts (d) and (e) follow from the characterization of $L(x)$ given above, since $\text{Ext}(T)$ is always Σ_1^1 and is Π_2^0 if P is bounded. $\square$

Next we turn to the definition of recursively continuous functions. For functions on ω^ω or $\{0,1\}^\omega$, a recursive function $y = F(x)$ is given by an oracle Turing machine which uses input x as an oracle to compute the values $y(n)$ and is continuous since each value $y(n)$ depends on only finitely many values of x .

Lemma 9.5 *A function $F : \omega^\omega \rightarrow \omega^\omega$ (respectively, $F : \{0,1\}^\omega \rightarrow \{0,1\}^\omega$) is recursively continuous if and only if there is a recursive function $f : \omega^{<\omega} \rightarrow \omega^{<\omega}$ (resp. $f : \{0,1\}^{<\omega} \rightarrow \{0,1\}^{<\omega}$) such that*

- (1) *for all $\sigma \prec \tau$, $f(\sigma) \preceq f(\tau)$,*
- (2) *for all $x \in \omega^\omega$, $\lim_{n \rightarrow \infty} |f(x \upharpoonright n)| = \infty$, and*
- (3) *for all $x \in \omega^\omega$, $\lim_{n \rightarrow \infty} f(x \upharpoonright n) = F(x)$.*

Proof. Given such a representation f for F , clearly we can compute $y(n)$ for $y = F(x)$ from x by computing $f(x \upharpoonright k)$ for sufficiently large k .

Given a recursive function F , define the representation f as follows. On inputting σ of length n , compute the values of $\tau(i)$ where $\tau = f(\sigma)$ for each $i < n$ by applying the algorithm for F for n steps, using oracle σ . The length of τ will be the least $k < n$ such that $\tau(k)$ does not converge in n steps. $\square$

In general, a function F on the Baire space is continuous if and only if it has a representation f as above. Thus F is continuous if and only if it is recursive in some parameter $x \in \omega^\omega$.

The definition of recursively continuous real functions is more difficult.

Definition 9.6 A function $F : [0,1] \rightarrow [0,1]$ is *recursively continuous* if there is a uniformly recursive sequence of functions $f_n : \mathcal{D} \rightarrow \mathcal{D}$ such that, for any $x \in \{0,1\}^\omega$, $F(r_x) = \lim_i f_i(q_{x \upharpoonright i})$ and a recursive function $\nu : \omega \rightarrow \omega$ such that, for all natural numbers m, n, k and all dyadic rationals q, r , if $|q - r| < 2^{-\nu(k)}$ and $m, n > \nu(k)$, then $|f_m(q) - f_n(r)| < 2^{-k}$.

This definition is easily seen to be equivalent to other standard definitions, such as those given by Lacombe [94]. See Pour-El-Richards [123] for some history.

Note for any recursively continuous real function, $F(x)$ is recursive real for any recursive real x .

Functions of several variables are treated similarly, thus a uniformly recursive sequence of functions $\{f_n\}_{n \in \omega}$ and a recursive function ν represent a continuous function $F : [0, 1]^2 \rightarrow [0, 1]$ if $\lim f_i(q_{x[i]}, q_{y[i]}) = F(x, y)$ for any reals x, y and if $|f_m(q_1, q_2) - f_n(r_1, r_2)| < 2^{-k}$ whenever $m, n > \nu(k)$ and both $|q_1 - r_1|, |q_2 - r_2| < 2^{-\nu(k)}$. For example, the standard distance function $|x - y|$ may be represented by taking $f_n(q, r) = |q - r|$ for all n and $\nu(k) = k + 1$.

We say a function $F : \{0, 1\}^\omega \rightarrow \{0, 1\}^\omega$ represents a real function G provided $y = F(x)$ whenever $r_y = G(r_x)$.

Lemma 9.7 *If F is a continuous (respectively recursive) map on $\{0, 1\}^\omega$ such that $F(x) = F(y)$ whenever $r_x = r_y$, then F represents a continuous (respectively recursively continuous) map on $[0, 1]$.*

Proof. Given the representation function f for F , let $f_i(q_\sigma) = q_{f(\sigma)}$ for all i and let $\nu(k)$ be the least n such that $|f(\sigma)| > k$ for all $\sigma \in \{0, 1\}^n$. $\square$

We remark that not every recursively continuous real function may be represented by a recursive function on $\{0, 1\}^\omega$; the distance function $|x - r|$ for any fixed rational $r \in (0, 1)$ is a counterexample. For example, suppose that $r = \frac{1}{6}$ and $G(x) = |x - \frac{1}{6}|$. Now suppose that $F : \{0, 1\}^\omega \rightarrow \{0, 1\}^\omega$ represents G and that $f : \{0, 1\}^{<\omega} \rightarrow \{0, 1\}^{<\omega}$ represents F . Now $G(\frac{2}{3}) = \frac{1}{2}$ which has two representations $x_1 = 1^\frown 0^\omega$ and $x_0 = 0^\frown 1^\omega$. Let $x_2 = (10)^\omega$ so that x_3 represents $\frac{2}{3}$. Then either $F(x_2) = x_1$ or $F(x_2) = x_0$. Suppose first that $F(x_2) = x_0$. Then for some n , $0 \prec f((10)^n)$. But then $(10)^n \frown 1^\omega$ is a number greater than $\frac{2}{3}$ so that $1 \prec f((10)^n 1^k)$ for some k which is a contradiction. Similarly if $F(x_2) = x_1$, then for some n , $1 \prec f((01)^n)$. But then $(10)^n \frown 0^\omega$ is a number less than $\frac{2}{3}$ so that $0 \prec f((10)^n 0^k)$ for some k which is again a contradiction.

A recursive metric on the Baire space is defined by $\delta(x, y) = 1/2^n = 0^n 1^\omega$, where n is the least such that $x(n) \neq y(n)$, and $\delta(x, y) = 0 = 0^\omega$ if $x = y$.

For a closed set K of a metric space (X, δ) , the distance $\delta_K(x)$ from a point x to the set K is defined to be $\min\{\delta(x, y) : y \in K\}$. It is well-known that δ_K is always a continuous function. We will see below that δ_K need not be recursively continuous even if K is a Π_1^0 class.

The graph of a function $F : X \rightarrow X$ is defined as usual to be $gr(F) = \{(x, F(x)) : x \in X\}$. For $X = \omega^\omega$, we can view the graph as a subset of X by associating the pair (x, y) with the element $z = x \otimes y$, where $z(2n) = x(n)$ and $z(2n+1) = y(n)$. For any class P and any $x \in X$, let $\pi_x(P) = \{y : x \otimes y \in P\}$.

For a function F from $[0, 1]$ to $[0, 1]$, the graph may be represented by a subset of $\{0, 1\}^\omega$, namely $\{x \otimes y : f(r_x) = r_y\}$.

A classical result says that a function on the interval is continuous if and only if the graph is closed. We give the effective version here.

Theorem 9.8

- (a) *The graph of a recursively continuous function on ω^ω is a Π_1^0 class.*
- (b) *Let X be either $\{0, 1\}^\omega$ or $[0, 1]$. Then a function $F : X \rightarrow X$ is recursively continuous if and only if the graph of F is a Π_1^0 class. Furthermore, the graph of any recursively continuous function may be represented by a tree with no dead ends.*

Proof.

(a) Suppose first that $F : \omega^\omega \rightarrow \omega^\omega$ is recursively continuous and is represented by $f : \omega^{<\omega} \rightarrow \omega^{<\omega}$. Define the recursive tree T with $[T] = gr(F)$ by putting $\sigma \otimes \tau \in T$ if and only if τ is consistent with $f(\sigma)$.

(b) Given a recursive $F : \{0, 1\}^\omega \rightarrow \{0, 1\}^\omega$, define the recursive tree T with $gr(F) = [T]$ as in (a). Then $Ext(T)$ is Σ_1^0 , and therefore recursive, by the following easily verified claim.

CLAIM: $\sigma \otimes \tau \in Ext(T) \iff (\exists \sigma' succ \sigma) \tau \prec f(\sigma')$.

Given a recursive tree T so that $gr(F) = [T]$, define the recursive representing function f by letting $f(\sigma)$ be the common part of $\{\tau : \sigma \otimes \tau \in T\}$.

Next suppose that F is a recursively continuous function on $[0, 1]$ and let the recursive sequence f_i of dyadic rational functions and the recursive modulus function ν be given as in Definition 9.6. We can assume that $\nu(k) > k$ for all k . In this case, we can define our desired recursive tree T with $gr(F) = [T]$ to be the set of pairs $\sigma \otimes \tau$ of length $2n - 1$ or $2n$ such that $|f_n(q_\sigma) - q_\tau| \leq 2^{1-k}$ for all k such that $|\sigma| \geq \nu(k)$. Again it is easy to see that $Ext(T)$ is recursive.

Suppose now that $gr(F)$ is a Π_1^0 class and, by Lemma 9.1, let $T \subseteq \{0, 1\}^{<\omega}$ be a recursive tree so that $[T]$ represents $gr(F)$. T may not be the graph of a function, since each dyadic real has two representations. However any two representations of length n differ by 2^{-n} . Thus, for any i and any σ of length n , we let $f_i(q_\sigma) = q_\tau$ for the lexicographically least τ of length $|\sigma|$ such that $\sigma \otimes \tau \in T$, and let $\nu(k)$ be the least n such that, for all σ of length n and any τ_1, τ_2 with $\sigma \otimes \tau_1$ and $\sigma \otimes \tau_2$ both in T , $\delta_q(\tau_1, \tau_2) < 2^{-k}$. $\square$

We next examine the complexity of the image of a Π_1^0 class under a recursively continuous function. The classical results are that the image of any compact set under a continuous function is compact and that the image of a closed set is an analytic set.

Theorem 9.9 *Let F be a recursively continuous function on a Π_1^0 subclass $P = [T]$ of ω^ω or $[0, 1]$, and let $F[P] = \{F(x) : x \in P\}$. Then*

- (a) $F[P]$ is a Σ_1^1 class,
- (b) if P is bounded, then $F[P]$ is a strong Π_2^0 class, and
- (c) if P is recursively bounded, then $F[P]$ is a recursively bounded Π_1^0 class and, furthermore, if there is a recursive tree T with no dead ends such that $P = [T]$, then there is a recursive tree S with no dead ends such that $F[P] = [S]$.

Proof.

- (a) This part follows immediately from the fact that

$$y \in F[P] \iff (\exists x)[x \in P \ \& \ \langle x, y \rangle \in gr(F)].$$

(b) Suppose that T is a finitely branching, recursive tree and let S be a recursive tree such that $gr(F) = [S]$. Then it follows from König's Lemma that $F[P] = [R]$, for the finitely branching Σ_1^0 tree R defined by

$$\tau \in R \iff (\exists \sigma)[\sigma \in T \ \& \ \sigma \otimes \tau \in S].$$

(c) Now suppose that T is recursively bounded and let F be represented by the recursive function $f : \omega^{<\omega} \rightarrow \omega^{<\omega}$. Then it is easy to see the definition above in (b) becomes recursive.

To find a bound for the possible value of $\tau(n)$ for $\tau \in R$, compute the least m such that $|f(\sigma)| > n$ for all $\sigma \in T$ of length m . Then we compute the maximum value $h(r)$ of $f(\sigma(n))$ for all $\sigma \in T$ of length n . Thus R is highly recursive.

If T has no dead ends, then it follows from the fact that $f(\sigma) \prec f(\rho)$ whenever $\sigma \prec \rho$ that R likewise has no dead ends.

For $P \subseteq [0, 1]$, the result now follows from Lemma 9.1 and Theorem 9.8. $\square$

Our next results may be viewed as partial converses of part (c) of Theorem 9.9.

Theorem 9.10

- (a) A Π_1^0 subclass P of $\{0,1\}^\omega$ is the recursively continuous image of $\{0,1\}^\omega$ if and only if it can be represented by a recursive tree with no dead ends.
- (b) A Π_1^0 subclass P of $[0,1]$ is the recursive image of $[0,1]$ if and only if $P = [r, s]$ for some recursive reals $r < s$.

Proof.

(a) Let $P = T$ where T is a recursive tree with no dead ends. Define the recursive map $F : \{0,1\}^\omega \rightarrow \{0,1\}^\omega$ so that $y = F(x)$ is some element of P which is nearest to x by letting $y(n) = x(n)$ as long as $x[n+1] \in T$ and letting $y(n+1)$ be the least i such that $y[n+1] \frown i \in T$ otherwise.

(b) If r and s are recursive reals, then $[r, s]$ is the image of $[0,1]$ under the recursively continuous map $F(x) = r + (s-r)x$.

Suppose that K is the image of the recursively continuous map F . It follows from the Intermediate Value Theorem that $K = [r, s]$ where the reals r and s are the maximum and minimum elements of P . It follows from Theorem 9.9 that K may be represented by a tree with no dead ends and then from Theorem 9.4 that r and s are recursive. $\square$

Corollary 9.11 *Let F be a recursively continuous function on ω^ω , $\{0,1\}^\omega$, or $[0,1]$. Then the maximum and minimum values of F on P are recursive reals (if they exist).*

Theorem 9.12 *Each of the following sets is a Π_1^0 class for any recursively continuous function $F : X \rightarrow X$, where the space X may be $\{0,1\}^\omega$, $[0,1]$, ω^ω or $\mathbb{R}$. In case (3), the class always has a recursive member when $X = [0,1]$. In case (4), the class is always bounded when $X = \mathbb{R}$.*

- (a) The set of points x where $F(x) = x_0$ for any fixed recursive x_0 .
- (b) The set of points where F attains its maximum (minimum).
- (c) The set of fixed points of F .
- (d) The Julia set of F where $X = \omega^\omega$ or $\mathbb{R}$.
- (e) The complement of the basin of attraction of a recursive periodic point.

Proof.

(a) This is immediate from Theorem 9.8.

(b) It follows from Corollary 9.11 that the maximum and minimum are recursive if they exist. The result now follows from part (b).

(c) This is easily reduced to part (b). For $\mathbb{R}$, x is a fixed point of F if and only if F is a zero of $G(x) = F(x) - x$. For $[0, 1]$, take $G(x) = |F(x) - x|$. For $\{0, 1\}^\omega$ or ω^ω , define $z = G(x)$ by $z(n) = |F(x)(n) - x(n)|$.

A recursive fixed point r may be found for a recursively continuous function on $X = [0, 1]$ by the standard procedure. If F has a dyadic fixed point, then there is nothing to do. If not, then repeatedly split the interval in two and choose the subinterval with $F(x) < x$ on one end and $F(x) > x$ on the other. Then r is the unique element in the intersection of these intervals.

(d) This is immediate from the characterization of the Julia set as $\{x : (\forall n) |F^n(x)| \leq c\}$ for a fixed recursive point c . Note that in ω^ω , $\{x : x \leq_L x_0\}$ is not a bounded Π_1^0 class in our sense of being the paths through a finite branching tree.

(e) Given an attracting point c for F , there is some recursive interval $(a, b) \subseteq B(x)$ containing c . Then the complement of the basin of attraction may be characterized as

$$\{x : (\forall n) (F^n(x) \leq a \vee F^n(x) \geq b)\}. \quad \square$$

As usual, we give a few immediate corollaries from the results of Section 2.

Theorem 9.13 *Let $F : X \rightarrow X$ be a recursively continuous map, where X is either $\{0, 1\}^\omega$, $[0, 1]$ or $\mathbb{R}$.*

- (a) *If F attains a maximum M , then there are two points x_1 and x_2 with $F(x_1) = F(x_2) = M$ such that any function recursive in both x_1 and x_2 is recursive.*
- (b) *If F has only countably many zeroes, then F has a recursive zero.*
- (c) *If F has only finitely many fixed points, then every fixed point of F is recursive.*
- (d) *If the Julia set of $F : \mathbb{R} \rightarrow \mathbb{R}$ has no recursive member, then it contains a continuum of pairwise Turing incomparable elements.*

- (e) *If the basin of attraction $B(x_0)$ of a recursive fixed point x_0 of F is not all of X , then there is a point x of r.e. degree which is not in $B(x_0)$.*

Proof. We just note that in each case, a function defined on $\mathbb{R}$ may be restricted to a finite interval and thus be treated as a map on the interval. For example, if F has a zero, take a recursive interval $[a, b]$ on F has a zero and let $[c, d]$ be the image of $[a, b]$ under F . Then F may be composed with maps between $[0, 1]$ and the two intervals to obtain a map $G : [0, 1] \rightarrow [0, 1]$ so that the set of zeroes of G is homeomorphic to a subset of the set of zeroes of F . $\square$

Theorem 9.14 *Let $F : \omega^\omega \rightarrow \omega^\omega$ be a recursively continuous map.*

- (a) *If F attains a maximum M , then $F(x) = M$ for a point x which is recursive in some Σ_1^1 set.*
- (b) *If F has only countably many zeroes, then F has a hyperarithmetic zero.*
- (c) *If F has only finitely many fixed points, then every zero of F is hyperarithmetic.*

Next we give the collection of converses to Theorem 9.13. The first three parts are due to Nerode and Huang [115] and may also be found in Ko [88].

Theorem 9.15 *Let P be a Π_1^0 subclass of the space X , either $\{0, 1\}^\omega$, $[0, 1]$, ω^ω or $\mathbb{R}$.*

- (1) *There is a recursively continuous function F such that P is the set of zeroes of F .*
- (2) *There is a recursively continuous function F with maximum value M such that $P = \{x : F(x) = M\}$.*
- (3) (a) *If X is either $\{0, 1\}^\omega$, ω^ω or $\mathbb{R}$, then there is a recursively continuous function F such that P is the set of fixed points of F .*
 (b) *If X is $[0, 1]$ and P has a recursive member, then there is a recursively continuous function F such that P is the set of fixed points of F .*

- (4) If P is bounded and has both a recursive maximum and a recursive minimum element, then there is a recursively continuous function such that
- (a) P is the complement of the basin of attraction of a recursive periodic point, where $X = [0, 1]$ or $\mathbb{R}$.
 - (b) P is the Julia set of F where $X = \mathbb{R}$.

Proof.

(1) First suppose $P \subseteq \omega^\omega$ and let T be a recursive tree such that $P = [T]$. Define the recursive function F by

$$F(x) = \begin{cases} 0^\omega, & \text{for } x \in P \\ 0^n \smallfrown 1 \smallfrown 0^\omega, & \text{if } n \text{ is the least with } x \restriction n \notin T. \end{cases}$$

If P represents a subset of $[0, 1]$, then the function F is modified when x represents a dyadic, so that $F(\sigma \smallfrown 1 \smallfrown 0^\omega) = F(\sigma \smallfrown 0 \smallfrown 1^\omega)$ for all σ . Thus when $r_x = r_y$ is dyadic, we let

$$F(x) = F(y) = \begin{cases} 0^\omega, & \text{for } x \in P \\ 0^n \smallfrown 1 \smallfrown 0^\omega, & \text{if } n \text{ is the least with } x \restriction n \notin T \text{ and } y \restriction n \notin T. \end{cases}$$

For a subset P of $\mathbb{R}$, let Q be the image of P under the isomorphism G with $(0, 1)$ together with the point 0, if P has no lower bound and the point 1, if P has no upper bound. Then let H be the recursively continuous map with set Q of zeroes. It follows that P is the set of zeroes of $H \circ G$.

(2) Let F be the function defined in the proof of (1) and observe that 0 is the minimum value of F in each case. For the maximum argument on $[0, 1]$ or $\mathbb{R}$, just take $G(x) = 1 - F(x)$. For the maximum argument on ω^ω , note that the range of F is a subset of $\{0, 1\}^\omega$ and take $G(x)(n) = 1 - F(x)(n)$.

(3) (a) Let F be given by (1) so that P is the set of zeroes of F . Now let $G(x) = F(x) + x$ for the real line, and, for ω^ω or $\{0, 1\}^\omega$, let $G(x)(n) = x(n)$ if $F(x)(n) = 0$ and $G(x)(n) = 1 - x(n)$, if $F(x)(n) \neq 0$.

(b) Let x_0 be a recursive member of the Π_1^0 class P and let F be the function given by (1) so that $x \in P$ if and only if $F(x) = 0$. Define $G(x)$ to be $x + (x_0 - x)F(x)$.

(4) (a) Let P be a Π_1^0 proper subclass of $[0, 1]$. Then there is some recursive element $x_0 \notin P$. Let F be the recursive function given by part (1) such that $F(x) = 0$ for $x \in P$ and $F(x) > 0$ for $x \notin P$. Let $P_1 = P \cap [0, x_0]$ and $P_2 = P \cap [x_0, 1]$. Let M_1 be the maximal element of P_1 and let M_2 be the minimal element of P_2 , so that both M_1 and M_2 are recursive. Now define the function G by cases.

$$\begin{aligned} G(x) &= M_1 + F(x)(x_0 - M_1), \text{ for } x \leq M_1, \\ G(x) &= x + (x - M_1)(x_0 - x), \text{ for } M_1 \leq x \leq x_0, \\ G(x) &= x - (M_2 - x)(x - x_0), \text{ for } x_0 \leq x \leq M_2, \\ G(x) &= M_2 - F(x)(M_2 - x_0), \text{ for } M_2 \leq x. \end{aligned}$$

Then x_0 , M_1 and M_2 are all fixed points of G . We claim that P is the complement of the basin of attraction of x_0 . The following inequalities are immediate from the above definition.

$$\begin{aligned} M_1 &\leq G(x) \leq x_0, \text{ for } x < M_1, \\ x &< G(x) < x_0, \text{ for } M_1 < x < x_0, \\ x_0 &< G(x) < x, \text{ for } x_0 < x < M_2, \\ x_0 &\leq G(x) < M_2, \text{ for } M_2 < x. \end{aligned}$$

First we show that the basin of attraction of x_0 for G includes $[M_1, M_2]$. Given $M_1 < x < x_0$, we see that $x < G(x) < x_0$. It follows that $G^n(x)$ is an increasing sequence with limit L such that $G(L) = L$ and $M_1 < L \leq x_0$. Thus we must have $L = x_0$. A similar argument works for $x_0 < x < M_1$.

Next suppose that $x \notin P$ and either $x < M_1$ or $x > M_2$. Then either $G(x) \in [M_1, M_2]$, so that x is in the basin of attraction of G .

Now suppose that $x \in P$, so that either $x \in P_0$ or $x \in P_1$. For $x \in P_0$, we have $F(x) = 0$ and $x \leq M_1$, so that $G(x) = M_1$ and thus $G^n(x) = M_1$ for all $n > 0$. Thus x is not in the basin of attraction of G . Similarly for $x \in P_1$, $G^n(x) = M_2$ for all $n > 0$, so that x is not in the basin of attraction of G .

For $X = \mathbb{R}$, just identify X with a subclass of $(0, 1)$ as in (1) above.

(b) Let P be a bounded Π_1^0 class of reals with a recursive minimal element m and a maximal element M and let the recursively continuous function F be given by (1) so that $F(x) = 0$ for all $x \in K$ and $F(x) > 0$ for all $x \notin K$. Now define the function F in the following cases.

$$\begin{aligned} G(x) &= m + M - x, \text{ for } x \leq m. \\ G(x) &= M + F(x), \text{ for } m \leq x \leq M. \\ G(x) &= 2x - M, \text{ for } x \geq M. \end{aligned}$$

□

Since any countable Π_1^0 subset of $[0, 1]$ and any Π_1^0 subset which may be represented by a tree with no dead ends has a recursive member, we have the following immediate corollary.

Corollary 9.16

- (a) *If the nonempty Π_1^0 subclass K of $[0, 1]$ may be represented by a tree with no dead ends, then K is the set of fixed points of some recursively continuous function from $[0, 1]$ into $[0, 1]$.*
- (b) *Any countable, nonempty Π_1^0 subclass K is the set of fixed points of some recursive function from $[0, 1]$ into $[0, 1]$.*

As usual, we have a number of immediate corollaries, of which we state only a few.

Theorem 9.17 *Let X be $\{0, 1\}^\omega$, ω^ω , $\mathbb{R}$, or $[0, 1]$.*

- (a) *For any r.e. degree $\mathbf{c}$, there is a recursively continuous function F on X such that the set of r.e. degrees which contain zeroes of F equals the set of r.e. degrees $\geq_T \mathbf{c}$.*
- (b) *There is a recursively continuous function F on X which has a fixed point and such that any two distinct fixed points are Turing incomparable if X is $\{0, 1\}^\omega$, ω^ω , or $\mathbb{R}$. There is a recursively continuous function F on $[0, 1]$ which has a unique recursive fixed point and uncountable many non-recursive fixed points and such that any two distinct non-recursive fixed points are Turing incomparable.*
- (c) *There is a recursively continuous function F which has a maximum M on X , such that there is a unique non-recursive point x_0 where M is attained and x_0 is also the unique accumulation point of the set where M is attained.*
- (d) *There is a recursively continuous function on $\mathbb{R}$ with an attracting point at infinity such that every recursive point is attracted to infinity but not every point is attracted to infinity.*
- (e) *There is a recursively continuous function on $[0, 1]$ with an attracting point at infinity such that every recursive point is attracted to infinity but not every point is attracted to infinity.*

Proof. Note that in part (b) when $X = [0, 1]$, we may add a single recursive point to the Π_1^0 class so that it can represent the set of fixed points. $\square$

Theorem 9.18

- (a) *There is a recursively continuous function on ω^ω which has a zero but has no hyperarithmetical zero.*
- (b) *There is a recursively continuous function on ω^ω which attains a maximum M such that $F(x) \neq M$ for any hyperarithmetical point x .*
- (c) *There is a recursively continuous function on ω^ω which has a fixed point but has no hyperarithmetical fixed point.*

Ko [89] recently improved part (4) of Theorem 10.15 by showing that if the Π_1^0 class P has either a p-time maximum element or a p-time minimum element, then there is a p-time computable function f with Julia set P . Furthermore, Ko shows in [89] that there is such a set P which has a non-recursive Hausdorff dimension, which implies that there is a p-time computable function f such that the Julia set of f has non-recursive Hausdorff dimension.

10 Gale-Stewart games and the Rado selection principle

These two problems fall under the category of miscellaneous combinatorial results and are thus united in one section. The set of winning strategies for an effective, closed $\{0, 1\}$ -game of perfect information was shown in [24] to strongly represent any r.b. Π_1^0 class. We will consider more general closed games here. The set of choice functions for an effective Rado family was shown in [70] to strongly represent any bounded Π_1^0 class. We will summarize those results below.

10.1 Gale-Stewart games

For any subset C of ω^ω , the infinite game $G(C)$ of perfect information is defined as follows. Two players, I and II, alternately play an infinite sequence $z = (x(0), y(0), x(1), y(1), \dots)$, and player II wins this play if $z \in C$.

A *strategy* for Player II is a (partial) function Θ from $\omega^{<\omega}$ into ω . For any play $x = (x(0), x(1), \dots)$ of the game by Player I, the play $\Theta(x)$ of the game when Θ is applied to x is given by $(x(0), y(0), x(1), \dots)$, where, for each n , $y(n) = \Theta((x(0), y(0), \dots, y(n-1), x(n)))$. The strategy Θ is said to be a *winning strategy* for Player II in the game $G(C)$ if, for any play x of the game by Player I, $\Theta(x) \in C$. The notion of a strategy and a winning strategy for Player I is similarly defined. The game $G(C)$ is said to be *determined* if one of the two players has a winning strategy. Gale and Stewart showed in [56] that the game $G(C)$ is determined if C is either closed or open. For a closed set C , we have $C = [T]$ for some tree T , and we will sometimes refer to $G(C)$ as $G(T)$. We say that $G(T)$ is a *recursively presented Gale-Stewart game* if T is a recursive tree and that $G(T)$ is *bounded* (respectively, *recursively bounded*) if the set $[T]$ is bounded (resp. r.b.).

As pointed out in [24], strategies need to be coded to avoid always having a perfect set of winning strategies.

Let $\tau_0, \tau_1, \dots$ effectively enumerate the nonempty elements of $\omega^{<\omega}$ in increasing order where we order the sequences by the sum of the sequence plus the length and then lexicographically. Thus $\tau_0 = (0)$, $\tau_1 = (00)$, $\tau_2 = (1)$, $\dots$. For each $\tau \in \omega^{<\omega}$, let $n(\tau)$ be the unique n such that $\tau = \tau_n$. Then an arbitrary sequence $z = (z(0), z(1), \dots) \in \omega^\omega$ codes a strategy Θ_z for Player II in the following manner. For any play $x = (x(0), x(1), \dots)$ of Player I, the strategy Θ_z produces the following response $y = (y(0), y(1), \dots)$ by Player II. First, $y(0) = z(n((x(0))))$ and for any k , $y(k+1) = z(n)$, where $\tau_n = (x(0), \dots, x(k))$, that is,

$$\Theta_z((x(0), y(0), \dots, y(k-1), x(k))) = z(\tau_n).$$

Thus $z(0) = \Theta_z((0))$, $z(1) = \Theta_z(0, \Theta_z(0), 0)$, $z(2) = \Theta_z((1))$, and so on. It is clear that the result $\Theta_z(x)$ of applying this strategy to a play $x = (x(0), x(1), \dots)$ of the game by Player I can be computed from x and z by a recursive function. For a finite sequence $z \upharpoonright n = (z(0), \dots, z(n-1))$, $\theta_{z \upharpoonright n}$ is a *partial strategy* which, applied to any partial play $x \upharpoonright m+1 = (x(0), \dots, x(m))$ of Player I with $n(x \upharpoonright m) < n$, gives a partial response $\theta_{z \upharpoonright n}((x(0), y(0), \dots, y(m-1), x(m))) = y(m)$ where for all $r \leq m$, $y(r) = z(k_r)$ if $n((x(0), \dots, x(r))) = k_r$.

Now, for any tree $T \subseteq \omega^{<\omega}$, let $\text{ws}(T)$ be the set of codes

$$z = (z(0), z(1), \dots) \in \omega^\omega$$

for winning strategies of Player II in the game $G(T)$.

Theorem 10.1 *For any recursive tree T ,*

- (a) $\text{ws}(T)$ is a Π_1^0 class.
- (b) *If T is finitely branching, then $\text{ws}(T)$ is bounded.*
- (c) *If T is highly recursive, then $\text{ws}(T)$ is recursively bounded.*

Proof. We will define a recursive tree Q such that $\text{ws}(T) = [Q]$, as follows. First $\emptyset$ is in Q and then for any $\sigma = (z(0), \dots, z(n-1))$, $\sigma \in Q$ if and only if, for all sequences $\nu = (x(0), \dots, x(r-1))$ where $n(\nu) < n$, the result of applying the partial strategy θ_σ coded by σ to the partial play ν is in T . It follows from the discussion above that there is a recursive function g such that, for each n , the value $z(n)$ of a coded strategy gives the play $y(g(n))$ of player II at step $g(n)$. If T is finitely branching, then there are only finitely many possible choices for $y(g(n))$ which allow player II to win the game, so that only finitely many values are possible for $z(n)$. This makes $\text{ws}(T)$ bounded. If T is highly recursive, then we can actually compute a list of these possible values from $g(n)$. Thus $\text{ws}(T)$ will be recursively bounded. $\square$

As usual, we can derive a number of immediate corollaries. We state the following and leave the rest to the reader.

Theorem 10.2 *Let T be a recursive tree such that player II has a winning strategy for the Gale-Stewart game $G(T)$.*

- (a) *There is a winning strategy which is recursive in some Σ_1^1 set and, if there are only finitely many winning strategies, then each winning strategy is hyperarithmetical.*
- (b) *If T is finitely branching, then there is a winning strategy which is recursive in $0''$.*
- (c) *If T is highly recursive, then there is a winning strategy of r.e. degree and, if there are only countably many winning strategies, then there is a recursive winning strategy.*
- (d) *If T is highly recursive and there is no recursive winning strategy, then there is a continuum of pairwise Turing incomparable winning strategies.*

Next we consider the set of winning strategies for Player I (who is trying to get the play into the open set). Let $WS'(T)$ be the set of codes for winning strategies of Player I. Note that for any recursively presented Gale-Stewart game $G(C)$, the set of winning strategies of Player I is a Σ_1^0 class.

Theorem 10.3 *For any recursive tree T ,*

- (a) $WS'(T)$ is a Π_1^1 class.
- (b) If T is finitely branching, then $WS'(T)$ is an open set.
- (c) If T is highly recursive, then $WS'(T)$ is a Σ_1^0 class.

Proof. We describe the class of actual strategies Θ and leave it to the reader to translate this into the coded strategies as in the proof of Theorem 10.1. In general, Θ is a winning strategy for Player I if and only if, for all plays y of Player II, the result $\Theta(y)$ of the game when Player I uses the strategy Θ is not in the set $[T]$, that is,

$$(\forall y)(\exists n)[(x(0), y(0), x(1), y(1), \dots, x(n), y(n)) \notin T]$$

where $x(i+1) = \Theta((x(0), y(0), \dots, x(i), y(i)))$ for all i .

If T is finitely branching, let $f(n)$ give an upper bound for the possible values of $\sigma(n)$ for any $\sigma \in T$. Then we can use König's Infinity Lemma as usual to express this in the form:

$$(\exists n)(\forall(y(0), y(1), \dots, y(n))) \\ [(x(0), y(0), x(1), y(1), \dots, x(n), y(n)) \notin T], \quad (*)$$

where each $y(i) \leq f(2i)$, so that the $(\forall)$ quantifier is bounded, which shows that $WS'(T)$ is an open set.

Finally, if T is highly recursive, then we may take the function f to be recursive, so that the characterization $(*)$ above makes $WS'(T)$ a Σ_1^0 class. $\square$

Theorem 10.4 *Let T be a recursive tree such that Player I has a winning strategy for the Gale-Stewart game $G(T)$.*

- (a) *There is a Δ_2^1 winning strategy and, if there are only finitely many winning strategies, then each winning strategy is Δ_2^1 .*
- (b) *If T is finitely branching, then there is a recursive winning strategy.*

Proof.

(a) This follows from the theorem that Δ_2^1 is a basis for Π_1^1 , which is a corollary of the Novikov-Kondo-Addison Uniformization Theorem (see Hinman [66, pp. 196–198] for details).

(b) Since $WS'(T)$ is open and nonempty, there must be an interval of coded winning strategies, which of course will contain a recursive strategy. $\square$

Now we consider the reverse direction of the correspondences given in Theorems 10.1 and 10.3.

Theorem 10.5 *For any recursive tree Q , there is a recursive tree T and an effective one-to-one degree preserving correspondence between the Π_1^0 class $[Q]$ of infinite paths through Q and the class $WS(T)$ of winning strategies for the effectively closed game $G(T)$. If Q is finitely branching (respectively highly recursive), then T may be taken to be finitely branching (resp. highly recursive).*

Proof. Let the recursive tree Q be given. Our basic idea is that each path $\Pi = (\pi(0), \pi(1), \dots) \in [Q]$ should correspond to a strategy Θ_Π which acts as follows. Given any partial play,

$$((x(0), \dots, x(m)))$$

of Player I, Θ_Π will respond with

$$\Theta_\Pi((x(0), y(0), \dots, y(m-1), x(m)) = y(m)$$

where $y(m) = 0$ if $x(i) > 0$ for any $i \leq m$ and $y(m) = \pi(m)$ if $x(i) = 0$ for all $i \leq m$. Thus whenever Player I plays a value $x(i) > 0$, then ever after Θ_Π will respond with a 0 and if Player I plays all 0's, then Θ_Π will respond by reproducing the path Π . It is easy to see that when we code the strategy Θ_Π via a sequence $z = (z(0), z(1), \dots)$ that z will have the same Turing degree as Π . Thus the correspondence $\Pi \rightarrow \Theta_\Pi$ will be an effective one-to-one degree preserving correspondence. Thus all we need to do is recursively define a recursive tree $T \subseteq \omega^{<\omega}$ so that

$$WS(T) = \{z : z \text{ is a code of } \Theta_\Pi \text{ for some } \Pi \in [Q]\}.$$

We begin with sequences (a, b) of length 2 by putting $(a, b) \in T$ if and only if, either $a > 0$ and $b = 0$ or $a = 0$ and $(b) \in Q$. (This ensures that if Player I

starts with an $x > 0$, then any winning strategy Θ for Player II must respond with a 0, whereas if Player I starts with a 0, then Player II must respond by starting a sequence in Q . Similar remarks will apply to the subsequent nodes we put in T .) Then, for each n and each

$$\tau = (x(0), y(0), \dots, x(n), y(n)) \in T,$$

do the following.

- (1) If $x(k) > 0$ for some $k \leq n$, then put $\tau \hat{\smallfrown} a \hat{\smallfrown} 0 \in T$ and leave $\tau \hat{\smallfrown} a \hat{\smallfrown} b$ out of T for all a and for all $b > 0$.
- (2) If $x(k) = 0$ for all $k \leq n$, then put $\tau \hat{\smallfrown} a \hat{\smallfrown} b \in T$ if and only if, either $a > 0$ and $b = 0$ or $a = 0$ and $(y(0), \dots, y(n), b) \in Q$.

It easily follows from the definition of T that for any $\Pi = (\tau(0), \pi(1), \dots) \in [Q]$, Θ_Π is a winning strategy for Player II for the game $G(T)$. Now suppose that Θ is a winning strategy for Player II for $G(T)$. Then we can define a $\Pi = (\pi(0), \pi(1), \dots) \in [Q]$ such that $\Theta = \Theta_\Pi$ by recursion as follows. For each n , let

$$\pi(n) = \Theta((0, \pi(0), 0, \pi(1), \dots, 0, \pi(n-1), 0)).$$

It is easy to see from our definition of T that $\Pi \in Q$ and that $\Theta = \Theta_\Pi$. Thus the correspondence $\Pi \rightarrow \Theta_\Pi$ is our desired effective one-to-one degree preserving correspondence between $[Q]$ and $\text{ws}(T)$.

Suppose now that Q is finitely branching (respectively, highly recursive). Let $f(\pi)$ be an upper bound on $\{s : \pi \hat{\smallfrown} s \in Q\}$; if Q is highly recursive, then f is recursive. Now given a partial code $\sigma = (z(0), \dots, z(n-1)) \in \text{ws}(T)$ for a strategy for the game $G(T)$, we will indicate how to compute an upper bound $g(\sigma)$ for $\{t : \sigma \hat{\smallfrown} t \in \text{ws}(T)\}$. First compute the n -th finite sequence $\tau_n = (\tau(0), \dots, \tau(k-1))$ in the enumeration described above, and use σ to compute the partial play

$$\pi = (\tau(0), y(0), \dots, \tau(k-2), y(k-2), \tau(k-1)).$$

This can be done since for any $i < k$, $\tau[i]$ appears before τ in the enumeration. Now there are two cases in the computation of $g(\sigma)$. If $\tau(k-1) > 0$, then $g(\sigma) = 0$ and if $\tau(k-1) = 0$, then $g(\sigma) = f(\pi)$. Thus $\text{ws}(T)$ is finitely branching and if Q is highly recursive, then g is recursive so that $\text{ws}(T)$ is highly recursive. $\square$

As usual, there are a number of immediate corollaries and we state only a few. Note that all of the examples below are games in which player II (who is trying to force the play into the closed set) has the winning strategy.

Corollary 10.6

- (a) *There is a recursively presented Gale-Stewart game such that Player II has a winning strategy but has no hyperarithmetical winning strategy.*
- (b) *There is a recursively presented, bounded Gale-Stewart game $G(C)$ such that Player II has a winning strategy and for any winning strategy Θ with $\mathbf{0}' <_T \Theta \leq_T \mathbf{0}''$, there is a Σ_2^0 set A such that $\mathbf{0}' <_T A \leq_T \Theta$.*
- (c) *For any r.e. degree $\mathbf{c}$, there is a recursively presented, recursively bounded Gale-Stewart game $G(C)$ such that Player II has a winning strategy and the set of r.e. degrees which contain winning strategies for $G(C)$ equals the set of r.e. degrees $\geq_T \mathbf{c}$.*

Next we consider the reverse direction for games in which Player I has a winning strategy. Here the bounded games all have recursive winning strategies and nothing more can be said. For the unbounded games, the reverse direction demonstrates the connection between Π_1^1 classes and the game quantifier of Moschovakis. Recall that the Π_1^1 class with index e is the set $[T_e]$ of infinite paths through the e -th primitive recursive tree T_e . A theorem of Moschovakis states that the set of indexes e such that Player I has a winning strategy for the game $G(T_e)$ is a universal Π_1^1 set. See [114] for a discussion of the game quantifier and this theorem.

Note that every winning strategy for Player I is a limit point of the set of winning strategies for Player I, since once the play of the game has gotten into the open set, Player I may play anything at all from that point on. Thus we cannot hope to represent even every Π_1^1 class with a one-to-one correspondence.

Theorem 10.7 *For any Π_1^1 class $Q \subseteq \omega^\omega$, there is a recursively presented Gale-Stewart game $G(C)$ and a recursive function F such that $y \in V \iff F(y) \in \text{ws}'(C)$.*

Proof. Suppose that $y \in Q \iff (\forall x)(\exists n) R(x[n, y[n]])$. Define the closed set C to be $\{(x, y) : (\forall n) \neg R(x[n, y[n]])\}$. For each $y \in \omega^\omega$, let $F(y)$ code the strategy which simply plays y in response to any play x of Player I. Then it is clear that $F(y)$ codes a winning strategy if and only if $y \in Q$. $\square$

Theorem 10.8

- (a) *There is a recursively presented Gale-Stewart game $G(C)$ such that the set $WS'(T)$ of winning strategies for Player I is not Σ_1^1 .*
- (b) *There is a recursively presented Gale-Stewart game $G(C)$ for which Player I has a winning strategy but has no hyperarithmetic winning strategy.*

Proof.

(a) This is immediate from Theorem 10.7.

(b) Let $Q = \{z\}$ be a Π_1^1 singleton such that z is not hyperarithmetic and let the game $G(C)$ and the recursive function F be given by Theorem 10.7. Then it is clear that Player I has a unique winning strategy which consists of playing $z(n)$ at his n -th turn, and that this strategy has the same degree as z . $\square$

10.2 The Rado Selection Principle

In this section, we summarize the results of Jockusch, Lewis and Remmel from [70]. A Rado Family consists of collection of finite subsets $\{A_i : i \in I\}$ of $A = \bigcup_{i \in I} A_i$ and a collection of finite partial functions

$$\{\varphi_F \in A^F : F \text{ is a finite subset of } I\}$$

such that for each finite subset F of I , $\varphi_F(i) \in A_i$ for all $i \in F$. The Rado selection problem is to find a choice function $f : I \rightarrow A$ such that for any finite subset F of I , there is a finite extension $E \supseteq F$ such that $f(i) = \varphi_E(i)$ for all $i \in F$. We call such a choice function a *Rado selector*. Rado proved in [125] that any such family has a Rado selector. A finite set $F = \{x_1 < \dots < x_n\}$ of natural numbers may be coded by

$$k = 2^{x_1} + 2^{x_2} + \dots + 2^{x_n}.$$

In this case, we write $F = D_k$. We let 0 code the empty set. Then a family $\{A_i : i < \omega\}$ of finite sets may be coded by a function f such that $A_i = D_{f(i)}$ for each i . Similarly a family of finite partial choice functions φ_F may be coded by a single function g such that $g(i) = j$ if and only if

$$D_j = \{2^{x+1}3^{y+1} : x \in D_i \text{ \& } \varphi_{D_i}(x) = y\}.$$

A Rado family together with the coding described above is an *effective Rado family* if $A = I = \omega$ and if the coding functions f and g are both recursive.

Given an effective Rado family $\mathcal{F}$ as above, let $Ch(\mathcal{F})$ be the set of functions $h : \omega \rightarrow \omega$ such that

- (i) $h(i) \in A_i$ for each i and
- (ii) for each finite $F \subseteq \omega$, there is a finite extension E such that $\varphi_E(i) = h(i)$ for all $i \in F$.

The following is Theorem 3 of [70].

Theorem 10.9 *For any effective Rado family $\mathcal{F}$, there is a bounded strong Π_2^0 class P and an effective, degree preserving correspondence between P and $Ch(\mathcal{F})$.*

Proof. We can define a tree T which is recursive in $\mathbf{0}'$ such that $[T] = Ch(\mathcal{F})$ as follows. A finite path $(y_0, y_1, \dots, y_n)$ is in T if and only if

- (i) $y_i \in A_i$ for all $i \leq n$ and
- (ii) there exists a finite set M such that $\{0, \dots, n\} \subseteq M$ and $\varphi_M(i) = y_i$ for all $i \leq n$. □

Applying Theorems 2.7, 2.14 and 2.24, we obtain the following.

Corollary 10.10 *Let $\mathcal{F}$ be an effective Rado family. Then*

- (a) $\mathcal{F}$ has a Rado selector of Σ_2^0 degree.
- (b) *If $\mathcal{F}$ has only finitely many Rado selectors, then $\mathcal{F}$ has a Rado selector which is recursive in $\mathbf{0}'$.*

The following is Theorem 2 of [70].

Theorem 10.11 *For any nonempty bounded strong Π_2^0 class P , there exists an effective Rado family $\mathcal{F}$ and an effective, degree preserving correspondence between P and $Ch(\mathcal{F})$.*

We can now apply the results of Section 1 to prove the following.

Corollary 10.12

- (i) *There is an effective Rado family such that, for any degree $\mathbf{a}$ of a Rado selector for $\mathcal{F}$ and any Σ_2^0 degree $\mathbf{b} \geq_T \mathbf{a}$, $\mathbf{b} = \mathbf{0}''$.*
- (ii) *There is an effective Rado family such that, for any two degrees $\mathbf{a}$, $\mathbf{b}$ of Rado selectors for $\mathcal{F}$, $\mathbf{a} \not\leq \mathbf{b} \vee \mathbf{0}'$.*
- (iii) *There is an effective Rado family such that, for any degree $\mathbf{a} \leq_T \mathbf{0}''$ of a Rado selector for $\mathcal{F}$, there is a Σ_2^0 degree $\mathbf{b}$ with $\mathbf{0}' \leq_T \mathbf{b} \leq_T \mathbf{a}$.*

11 Feasible Π_1^0 Classes and Structures

In this section, we prove a strengthening of a result of Grigorieff [61] and Cenzer and Remmel [21] that every relational structure is recursively isomorphic to a feasible structure. Our result will imply that for essentially all the recursive mathematical problem of Sections 3–10, if a Π_1^0 class P can be represented as the set of solutions to a given recursive instance of the mathematical problem, then it can also be represented as the set of solutions to a polynomial time instance of the problem.

We also define the notion of a feasible tree and a feasible Π_1^0 class and show that every recursively bounded Π_1^0 class P is the set of infinite paths through a p-time tree. We consider the problem of whether a feasible Π_1^0 class has a feasible infinite path and give some p-time versions of the classical results from Section 2. These results will be applied in the next section to give conditions under which certain feasible mathematical problems will have feasible solutions.

There are two approaches that we can take to the study of feasible combinatorial problems. The first approach produces negative results, that is, feasible problems without feasible solutions. This approach is based on the observation that each of the problems considered arises from certain simple extensions of recursive relational structures. The solution set of such problems is then essentially determined by the underlying recursive relational structure. Thus we are led to the question of whether an arbitrary recursive structure is recursively isomorphic to a polynomial time structure. The fundamental theorem of Grigorieff [61] on feasible structures, is that every relational structure with finitely many relations is recursively isomorphic to a real-time structure. Cenzer and Remmel [21] extended this to show

that every recursive relational structure is recursively isomorphic to a p-time structure. We will apply this result to show that almost all the recursive combinatorial problems considered in this paper are recursively isomorphic to a feasible problem. Such an analysis was done by Cenzer and Remmel [22] for the feasible graph coloring problem. It was shown that every recursive graph is recursively isomorphic to a polynomial-time (p-time) graph, which implies that every r.b. Π_1^0 class can be represented as the set of colorings of a p-time graph, so that there are p-time 3-colorable graphs which have no recursive 3-colorings. Similarly, it is shown in [22] that every r.b. Π_1^0 class can be represented by the set of winning strategies for a feasibly closed game, so that there is a feasibly closed game with no recursive winning strategy.

One particular critical difference between feasible structures and recursive structures is the following. Any infinite recursive structure, such as a graph, is recursively isomorphic to a recursive structure with universe ω . This is because any infinite recursive subset of ω is recursively isomorphic to ω . However, it is not the case that every infinite feasible subset of ω is feasibly isomorphic to ω . Thus the fundamental result mentioned above provides, for every relational structure, a recursively isomorphic feasible structure which has for its universe a feasible subset of ω . It is not generally possible to make the universe all of ω . An example is given in [21] of a recursive linear ordering which is not recursively isomorphic to any p-time linear ordering on ω .

The second approach produces positive results and is based on the following idea. As we have indicated, the set of solutions to a combinatorial problem can be presented as the recursively bounded Π_1^0 class $[T]$ of infinite paths through a recursive tree T . Cenzer and Remmel introduced in [24] the notion of a feasible tree and showed that for any recursive tree T , there is a feasible tree T' such that $[T] = [T']$. This seems to indicate that feasibly Π_1^0 classes might be no better than arbitrary Π_1^0 classes. However, the key to whether a recursive problem has a recursive solution lies in the notion of a recursively bounded problem and the corresponding notion of a recursively bounded Π_1^0 class. Feasible versions of these notions are studied in [24] and used to obtain some general conditions which imply the existence of feasible solutions to feasible problems. Two positive results about the r.b. Π_1^0 class $P = [T]$ of infinite paths through a recursive tree T are the following.

- (1) *If T has no dead ends, that is, if every finite path in T can be extended to an infinite path, then P has a recursive member.*
- (2) *P is finite, then every member of P is recursive.*

The precise notions of feasible functions, graphs and trees depend strongly on the specific representation of the functions, graphs and trees as well as on the specific representation of the natural numbers. Several versions of feasibility are defined in [24] and used to obtain feasible versions of (1) and (2). These results are applied in [22], where it is shown that the set of colorings of a p -time graph $\mathcal{G}$ may be represented by a p -time tree. This implies, in particular, the following result.

Suppose that G is a k -colorable graph which is p -time in tally and that any k -coloring of a finite subgraph of G has an extension to a k -coloring of G . Then G has a p -time k -coloring in tally, and an exponential time k -coloring in binary. If, in addition, G has only finitely many k -colorings, then every k -coloring of G is p -time in tally and exponential time in binary.

A vertex set $V \subset \omega$ will have a tally representation given by $\text{tal}(V) = \{\text{tal}(n) : n \in V\}$ and a binary representation $\text{bin}(V) = \{\text{bin}(n) : n \in V\}$. Similarly, a set E of edges on V will have a tally representation given by $\text{tal}(E) = \{[\text{tal}(m), \text{tal}(n)] : (m, n) \in E\}$ and a binary representation $\text{bin}(E) = \{[\text{bin}(m), \text{bin}(n)] : (m, n) \in E\}$. For a graph $G = (V, E)$ such that $V \subseteq \omega$, we then let $\text{tal}(G) = (\text{tal}(V), \text{tal}(E))$ and let $\text{bin}(G) = (\text{bin}(V), \text{bin}(E))$. Then, for any notion of feasibility, we say that G is *feasible in binary* if $\text{bin}(G)$ is feasible and we say that G is *feasible in tally* if $\text{tal}(G)$ is feasible. A similar result is given in [24] for feasibly closed games and feasible winning strategies.

A relational structure is simply a structure which has no functions. We will present an improved version of the theorem from [21] that every recursive relational structure is recursively isomorphic to a polynomial time structure. This theorem will be our primary tool in the analysis of recursive combinatorial structures. It is important to note that the polynomial time structure provided will have for its universe a polynomial-time set possibly different from $\{1\}^*$ or $\{0, 1\}^*$. An example is constructed in [21] which shows that the theorem fails if any fixed polynomial time set A is specified in advance as the universe of the structure. The improved version of the theorem presented here applies to structures with two distinct types of objects, the first type being the normal universe of the structure, and with functions which map the first type into the second type. The type of example that we have in mind is a function from the vertices of a graph into the natural numbers

which computes the degree of a vertex. The universe of the graph is now expanded by adding a p -time set which represents the natural numbers and the degree function now becomes part of the structure. Naturally, the new objects are not vertices and therefore are not joined to any other objects by edges.

Theorem 11.1 *Let*

$$\mathcal{C} = (C, A, B, \{R_i^C\}_{i \in S}, \{f_i^C\}_{i \in T}),$$

be a recursive structure such that:

- (i) *A and B are disjoint subsets of C with $C = A \cup B$, and B is a polynomial time set.*
- (ii) *there is a recursive isomorphism from $\text{Bin}(\omega)$ onto a subset of $\text{Bin}(\omega) \setminus B$ with a p -time inverse.*
- (iii) *for each $i \in T$, f_i maps C into B .*
- (iv) *for each $i \in S$, the relation R_i is independent of B , that is, for any $(x_1, \dots, x_n) \in C^n$, where $n = s(i)$, any $j \leq n$ such that $x_i \in B$, and any $b \in B$,*

$$R_i^C(x_1, \dots, x_n) \text{ if and only if } R_i^C(x_1, \dots, x_{j-1}, b, x_{j+1}, \dots, x_n).$$

- (v) *for each $i \in T$, the function f_i is independent of B , that is, for any $(x_1, \dots, x_n) \in C^n$, where $n = t(i)$, any $j \leq n$ such that $x_i \in B$, and any $b \in B$,*

$$f_i^C(x_1, \dots, x_n) = f_i^C(x_1, \dots, x_{j-1}, b, x_{j+1}, \dots, x_n).$$

Then there is a recursive isomorphism φ of $\mathcal{C}$ onto a p -time structure $\mathcal{M}$ such that $\varphi(b) = b$ for all $b \in B$.

Proof. The idea of the proof is that we will replace each element x of A by a string y which codes x and is long enough to allow us to compute whether $x \in A$ in time $|y|$ and also to compute the relations and functions on A in time $|y|$ for all inputs which are less than or equal to x . These new strings may accidentally be in the set B , which must be kept disjoint from $A^{\mathcal{M}}$. This

is the reason for the p-time mapping which takes an arbitrary string to one which is not in B . Let ψ be a p-time map from $\text{Bin}(\omega)$ into $\text{Bin}(\omega) \setminus B$ such that ψ^{-1} is also p-time. We can assume that A is an infinite set, since, if A is finite, then $\mathcal{C}$ is p-time itself. Let $\sigma_0, \sigma_1, \dots$ be an effective enumeration of A in the usual order. Let b_0 be the shortest element of B . For any $x \in A$, we let $\nu(x)$ denote the number of steps needed to run the following algorithm.

First start to list $\sigma_0, \sigma_1, \dots$ until we find an s such that $\sigma_s = x$. Next for each $i \leq s$ such that $i \in S \cup T$, list all sequences $(x_1, \dots, x_n)$ from $\{b_0, \sigma_0, \dots, \sigma_s\}^n$ for $n = s(i)$ or $t(i)$ and then, for $i \in S$, compute whether $R_i(x_1, \dots, x_n)$ holds and, for $i \in T$, compute $f_i^{\mathcal{C}}(x_1, \dots, x_n)$.

Observe that the algorithm is completely uniform in x because our definition of recursive structure ensures that there is a recursive relation R such that $R(i, \langle x_1, \dots, x_{t(i)} \rangle) \iff R_i(x_1, \dots, x_{t(i)})$ and a recursive function f such that $f(i, \langle x_1, \dots, x_{t(i)} \rangle) = f_i(x_1, \dots, x_{t(i)})$. Note that in order to obtain the list $\sigma_0, \dots, \sigma_s$, we have to test whether $a \in A$ for all $a \leq x$. We then define a structure

$$\mathcal{M} = (M, \{R_i^{\mathcal{M}}\}_{i \in S}, \{f_i^{\mathcal{M}}\}_{i \in T})$$

as follows. For each $a \in A$, let $\varphi(a) = \psi(a \smallfrown 0^{1^{\nu(a)}})$ and, for each $b \in B$, we let $\varphi(b) = b$. It is clear that φ is a recursive isomorphism from $\mathcal{C}$ onto a subset M of $\text{Bin}(\omega)$, that $\varphi(B) = B$ and that $\varphi(A)$ is disjoint from B . The structure $\mathcal{M}$ is the image of $\mathcal{C}$ under the isomorphism φ . This means that $A^{\mathcal{M}} = \{\varphi(a) : a \in A\}$, $B^{\mathcal{M}} = B$, and $M = A^{\mathcal{M}} \cup B^{\mathcal{M}}$. For each $i \in S$ and $(x_1, \dots, x_n) \in C$, $R_i^{\mathcal{M}}$ is defined by

$$R_i^{\mathcal{M}}(\varphi(x_1), \dots, \varphi(x_n)) \iff R_i^A(x_1, \dots, x_n),$$

where $s(i) = n$. For each $i \in T$, $f_i^{\mathcal{M}}$ is defined by

$$f_i^{\mathcal{M}}(\varphi(x_1), \dots, \varphi(x_n)) = \varphi(f_i^A(x_1, \dots, x_n)),$$

where $t(i) = n$.

It is clear that the function φ is a recursive isomorphism from $\mathcal{A}$ onto $\mathcal{M}$. It remains to be seen that $\mathcal{M}$ is a polynomial time structure, that is, that M is a polynomial time set and that each relation $R^{\mathcal{M}}$ and function $f^{\mathcal{M}}$ is p-time.

We show that M is p-time as follows. It clearly suffices to show that $A^{\mathcal{M}}$ is p-time, since $B^{\mathcal{M}} = B$ is p-time. The procedure for testing whether an input y is in $A^{\mathcal{M}}$ is to compute $\psi^{-1}(y)$, check to make sure that it has a 0 in it, and then determine x and n such that $\psi^{-1}(y) = x \smallfrown 0 \smallfrown 1^n$. Then we simply run the algorithm outlined above to input x for n steps. Then $y \in A^{\mathcal{M}}$ if and only if the algorithm terminates in exactly n steps and gives the answer that $x \in A$.

We show that the function $f_i^{\mathcal{M}}$ is p-time as follows. Fix i and let $f = f_i$, let $n = t(i)$ and let c be the maximum amount of time required to compute $f^C(x_1, \dots, x_n)$ when $\{x_1, \dots, x_n\} \subseteq \{b_0, \sigma_0, \sigma_1, \dots, \sigma_{i-1}\}$. Now given input $(y_1, \dots, y_n)$, where each $y_i \in M$, the procedure for computing $f^{\mathcal{M}}(y_1, \dots, y_n)$ is the following.

First replace every $x_i \in B$ with $x'_i = b_0$ and let $x'_i = x_i$ for $x_i \in A$. Then compute $f^C(x'_1, \dots, x'_n)$. We claim that this computation takes time at most $c + \max\{|y_j| : 1 \leq j \leq n\}$. There are two cases of this claim to consider. First, if $\{x'_1, \dots, x'_n\}$ is a subset of $\{b_0, \sigma_0, \dots, \sigma_{i-1}\}$, then, by the definition of c , the computation takes at most c steps. On the other hand, if at least one of the $x'_j = x_j = \sigma_s$ for some $s \geq i$, then by the definition of ν , the computation takes less than $\nu(x_j)$ steps for some j ; but of course $\nu(x_j) < |y_j| \leq \max\{|y_j| : 1 \leq j \leq n\}$.

The argument for the relations is similar. This completes the proof of Theorem 11.1. $\square$

It follows from the proof that $\mathcal{M}$ is actually a linear-time structure if B is a linear time set. Note that since the set B is preserved under the isomorphism, we can carry along any p-time relations and functions that apply only to B . Also, if the set B is finite, then we can omit conditions (iv) and (v) from the hypothesis and modify the proof by adding all of B to the initial segment of elements of A used in the algorithm which defines the function ν .

Since the set of solutions of the various mathematical problems that we have considered correspond to the set of paths through some recursive tree, we are interested in the possible complexity of recursive trees and of the paths through those trees.

The *binary representation* $\text{bin}(\sigma)$ of a sequence $\sigma = (n_0, \dots, n_{k-1})$ is defined to be $(\text{bin}(n_0), \dots, \text{bin}(n_{k-1}))$ and the *binary representation* $\text{bin}(T)$ of a tree T is defined to be the set of finite strings $\{\text{bin}(\sigma) : \sigma \in T\}$. Then

we say that T is *p-time in binary* if $\text{bin}(T)$ is a polynomial time subset over the alphabet Σ consisting of 0 and 1 together with left and right parentheses and a comma. Given an infinite path $x = (x(0), x(1), \dots) \in \omega^\omega$, the binary representation of x is the function $\text{bin}(x)$ from $\text{Tal}(\omega)$ to $\text{Bin}(\omega)$ defined by $\text{bin}(x)(\text{tal}(i)) = \text{bin}(x(i))$. Then the path x is said to be *polynomial time in binary* if the function $\text{bin}(x)$ is the restriction of a p-time function from $\{0, 1\}^*$ to $\{0, 1\}$. Similar definitions can be given for the tally representations $\text{tal}(T)$ of T and $\text{tal}(x)$ of x as well as for other notions of feasibility, such as exponential time and non-deterministic polynomial time (NP). See [23] for details. In particular, note that if T is p-time in binary, then T is p-time in tally, since $\text{bin}(n)$ can be computed from $\text{tal}(n)$. On the other hand, since both $\text{bin}(x)$ and $\text{tal}(x)$ take tally representations as input, if the path x is p-time in tally, then x is also p-time in binary.

Next we consider feasible versions of the notion of a highly recursive tree. Define the function h with domain T by letting $h(\sigma)$ be the sequence $(i_1, \dots, i_d)$ which lists in increasing order the numbers i such that $\sigma \frown i \in T$. Then T is highly recursive if and only if h is recursive. The binary representation $\text{bin}(h)$ is defined to map $\text{bin}(\sigma)$ to $(\text{bin}(i_1), \dots, \text{bin}(i_d))$. The p-time tree T is said to be *locally p-time in binary* if the function $\text{bin}(h)$ is p-time. A tree T is said to be *p-time bounded in binary* if there is a p-time function θ such that, for all natural numbers k and all $\sigma = (n_0, \dots, n_k) \in T$, $|\text{bin}(n_k)| \leq p(1^k)$. Finally, T is said to be *highly p-time in binary* if T is p-time, locally p-time and p-time bounded in binary. Similar definitions can be given for the notion of p-time in tally and for other notions of feasibility. In particular, if T is p-time bounded in tally, then T is p-time bounded in binary. On the other hand, the notions of locally p-time in binary and tally are independent.

Our next theorem shows that any Π_1^0 class can be realized as the set of infinite paths through a p-time tree.

Theorem 11.2 *Let T be a recursive tree. Then there is a polynomial time tree P such that $[T] = [P]$. Furthermore, if T is recursively bounded, then P is also recursively bounded and if T is p-time bounded, then P is also p-time bounded.*

Proof. The same argument works for both the binary and the tally representations. We will give the binary argument for the first part and the tally

argument for the second part, since these are the stronger results. Let φ be a recursive function from $\omega^{<\omega}$ into $\{0, 1\}$ such that $\sigma \in \text{bin}(T) \iff \varphi(\sigma) = 1$. Let φ^s denote the partial recursive function which results by computing φ for exactly s steps on any input and let T^s be the s -th approximation to T , given by

$$\sigma \in T^s \iff \varphi^s(\text{bin}(\sigma)) = 1 \text{ or is undefined.}$$

Thus $T^0 \supset T^1 \supset \dots$ and, for any σ , $\sigma \in T \iff (\forall s)(\sigma \in T^s)$.

Now define the p-time tree P by letting

$$\sigma \in P \iff (\forall \tau \prec \sigma) \tau \in T^{|\text{bin}(\sigma)|}.$$

Note that P is a p-time tree in binary since to compute whether $\tau \in T^{|\text{bin}(\sigma)|}$ requires $|\text{bin}(\sigma)|$ steps for all τ so that to compute whether $\sigma \in P$ requires roughly $|\text{bin}(\sigma)|(|\text{bin}(\sigma)| + 1)$ steps.

It follows from the definition of P that $T \subseteq P$, so that $[T] \subseteq [P]$. Now suppose that $x \notin [T]$. Then there is some initial segment $\tau = x \upharpoonright n$ which is not in T . This means that, for some s , $\tau \notin T^s$. Since the sequence T^s is decreasing, we may assume that $s > n$. Now let $\sigma = x \upharpoonright s$, so that $|\text{bin}(\sigma)| \geq s$. It follows from the definition of P that $\sigma \notin P$. This implies that $x \notin [P]$. Thus $[T] = [P]$.

Now suppose that T is recursively bounded in tally and let p be the recursive function which computes, for each k , an upper bound $p(1^k)$ (in tally) for the possible value of n_k for any node $\sigma = (n_0, \dots, n_k) \in T$.

Suppose first that p is actually p-time. Then we can recursively define a tree Q such that $T \subseteq Q \subseteq P$ by putting $\sigma = (n_0, \dots, n_k) \in Q$ if and only if $\sigma \in P$ and, for all $i \leq k$, $n_i \leq p(1^i)$. It is clear that $[Q] = [T]$ and that Q is p-time since P and p are p-time.

Finally, suppose only that p is recursive and let p^s be the usual result of computing p for s steps. Once again we can define a highly recursive tree Q such that $T \subseteq Q \subseteq P$ by putting $\sigma = (n_0, \dots, n_k) \in Q$ if and only if $\sigma \in P$ and, for all $i \leq k$, either $p^k(1^i)$ is undefined or $n_i \leq p^k(1^i)$. Then again it is easy to check that Q is p-time in binary and that $[Q] = [T]$. $\square$

This result can now be combined with results from Section 2. Let us say that a Π_1^0 class P is *p-time presented* if $P = [T]$ for some polynomial-time tree T and that P is *p-time bounded* (p.b.) if $P = [T]$ for a p-time bounded tree T .

Corollary 11.3 *There is a nonempty p -time presented Π_1^0 class P such that P has no Δ_1^1 members.*

Corollary 11.4

- (a) *There is a p.b. Π_1^0 class P such that for any degree $\mathbf{a}$ of a member of P and any r.e. degree $\mathbf{b} \geq_T \mathbf{a}$, $\mathbf{b} = \mathbf{0}'$.*
- (b) *For any r.e. degree $\mathbf{c}$, there is a p.b. Π_1^0 class P such that the r.e. degrees of members of P are exactly those $\geq_T \mathbf{c}$.*
- (c) *For any degree $\mathbf{a}$, there is a p.b. Π_1^0 class P with no members of either degree $\mathbf{a}$ or degree $\mathbf{0}$.*
- (d) *There is a p.b. Π_1^0 class P such that any two members of P are Turing incomparable.*

Recall from Section 2 that if the tree T has no dead ends, then T has a recursive infinite path and that if T has only finitely many infinite paths, then each of those paths is recursive.

We will show below in Theorem 12.6 that the p -time versions of these results fail to be true. However, there are modified versions of both results. The following is Theorem 3.7 of [24].

Theorem 11.5

- (a) *Let $\text{Ext}(T)$ be a locally p -time tree in tally (respectively binary) and let $[T]$ be nonempty. Then $[T]$ contains an infinite path which is double exponential time computable in tally (resp. binary). Furthermore, if $\text{Ext}(T)$ is locally p -time in tally (resp. binary) and $[T]$ is finite, then every element of $[T]$ is computable in double exponential time in tally (resp. binary).*
- (b) *Let $\text{Ext}(T)$ be a locally p -time tree in tally (respectively binary) and let $[T]$ be nonempty. Moreover, assume that there is a linear time function h such that for all $\sigma = (n_0, \dots, n_k) \in T$, $h(b(\sigma))$ lists all $b(n)$ such that $(n_0, \dots, n_k, n) \in T$ where $b(\) = \text{tal}(\)$ if T is p -time in tally and $b(\) = \text{bin}(\)$ if T is p -time in binary. Then $[T]$ contains an infinite path which is exponential time computable in tally (resp. binary). Furthermore, if $[T]$ is finite, then every element of $[T]$ is computable in exponential time in tally (resp. binary).*

- (c) If $\text{Ext}(T)$ is a highly p -time tree in tally (resp. binary) and $[T]$ is nonempty, then $[T]$ contains an infinite path which is p -time time in tally (resp. binary). Furthermore, if $[T]$ is finite, then every element of $[T]$ is p -time in tally (resp. binary).
- (d) If $\text{Ext}(T)$ is a p -time bounded, p -time tree in binary and $[T]$ is nonempty, then $[T]$ contains an infinite path which is EXPTIME in binary. Furthermore, if $[T]$ is finite, then every element of $[T]$ is NP in binary.

We can now sharpen the contrast between recursive and p -time trees.

Theorem 11.6

- (a) There is a $p.b.$ Π_1^0 class P with a unique element x which is not p -time.
- (b) There is a highly recursive tree T with no dead ends such that there is no highly p -time tree S without dead ends such that $[T] = [S]$.

Proof. Let x be recursive but not p -time and let $P = \{x\}$. Then P is an r.b. Π_1^0 class, so it follows from Theorem 11.2 that $P = [S]$ for some p -time bounded, p -time tree S . This establishes (a).

On the other hand, $P = [T]$ where $T = \{(x(0), \dots, x(n-1)) : n < \omega\}$ is highly recursive and has no dead ends. Now suppose that S were highly p -time without dead ends and that $[S] = [T] = \{x\}$. Then it would follow from Theorem 12.5 (c) that x is p -time. This contradiction establishes (b). $\square$

12 Feasible versions of combinatorial problems

The main goal in this section is to apply the results of Section 11 to the mathematical problems discussed in Sections 2 through 9.

We observe that any feasible structure is recursive, therefore the set of solutions to a feasible problem is also the set of solutions to a recursive problem. Thus results such as Theorems 5.1, 6.1 and 7.1 have feasible versions. The reverse direction is more interesting.

We consider recursive representation theorems such as Theorems 5.3, 6.3 and 7.3, and corollary results such as Theorems 5.4, 6.4 and 7.4. These representation theorems showed that the set of solutions to a recursive problem of

various sorts can represent either every r.b. Π_1^0 class or at least every Π_1^0 class of separating sets. In this section we obtain better results, in most cases, by improving “recursive” to “polynomial-time”. Now an infinite recursive problem may be assumed to have universe ω , since any two infinite recursive sets are recursively isomorphic. (Here the universe of a graph-coloring problem, for example, is the set the vertices.) However, it is not true that any two polynomial-time sets are polynomial-time isomorphic. (For example, it is clear that there is no p-time map from $Tal(\omega)$ onto $Bin(\omega)$.) Thus a polynomial-time structure with some p-time set for its universe may not be recursively isomorphic to any p-time structure with universe ω . For example, a p-time Abelian groups with all elements of finite order is constructed by Cenzer and Remmel in [23] which is not even isomorphic to any p-time group with standard universe either ($Tal(\omega)$ or $Bin(\omega)$). For most of the problems considered above, we will show that any recursive problem can be reduced first to a p-time problem and then to a p-time problem with standard universe.

We illustrate the general strategy with the graph coloring problem. Recall from Section 6 that, for $k \geq 3$, the set of k -colorings of a recursive graph can be represented by an r.b. Π_1^0 class, and conversely can represent an arbitrary r.b. Π_1^0 class. Let $G = (V, E)$ be a recursive graph. Then the set of k -colorings of G can be represented as the Π_1^0 class $[T]$ of infinite paths through a recursive tree T . Now Theorem 11.2 constructs for us a p-time tree P such that $[T] = [P]$. Then the converse representation creates from P a graph whose k -colorings are in an effective degree preserving finite-to-one correspondence with the infinite paths through P . Furthermore, inspection of the proof from [129] shows that this graph will actually be polynomial time, since P is polynomial time. This shows that the k -colorings of any recursive graph can always be placed in an effective degree preserving correspondence with the k -colorings of some p-time graph, and, therefore, that the k -colorings of a p-time graph can strongly represent any r.b. Π_1^0 class.

However, there is no natural correspondence between the recursive graph and the p-time graph constructed in this manner. We can do better using Theorem 11.1.

Theorem 12.1 *For each recursive instance P of any of the following problems, there is a p-time instance Q of the problem which is recursively isomorphic to P . Furthermore, except in cases (13) and (14), if P has a recursive solution, then we can take Q to have a p-time solution.*

- (1) *Finding a k -coloring for a k -colorable highly recursive graph, for any $k \geq 3$.*
- (2) *Finding a marriage in a highly recursive society.*
- (3) *Finding a surjective marriage in a symmetrically highly recursive society.*
- (4) *Finding a surjective marriage in a symmetrically highly recursive society where each person knows at most two other people.*
- (5) *Finding a k -partition of a highly recursive graph such that no set in the partition is adjacent to m other sets, for $m > 2$.*
- (6) *Finding a one-way (or two-way) Hamiltonian (or Euler) path starting from a fixed vertex for a highly recursive graph.*
- (7) *Covering a recursive poset of width k by k chains, for any $k \geq 2$.*
- (8) *Covering a recursive poset of height k by k antichains, for any $k \geq 2$.*
- (9) *Expressing a recursive partial ordering on a set as the intersection of d linear orderings on the set.*
- (10) *Finding a subordering of type ω (or of type ω^*) of a recursive ordering.*
- (11) *Finding an ω -successivity (or an ω^* -successivity) in a recursive linear ordering.*
- (12) *Finding a non-trivial self-embedding of a recursive linear ordering.*
- (13) *Finding a winning strategy for an effectively closed binary game.*
- (14) *Finding a prime ideal of a recursive Boolean algebra.*

Proof. For problems (1) through (12), this follows immediately from Theorem 11.1, since each of these problems can be viewed as a relational structure and the given solution can be viewed as a function mapping to a fixed range. In the dimension of posets problem, we can interpret the solution as a finite set of relations. For problem (13), Theorem 4.4 of [24] shows that any recursive game may viewed as a p -time game in that the set of infinite paths which are winning for Player I will be the set of infinite paths through a p -time tree. For problem (14), Theorem 2.6 of [21] shows that any recursive Boolean algebra is recursively isomorphic to a p -time Boolean algebra. $\square$

We note that a recursive game with a recursive winning strategy is not necessarily isomorphic to a p-time game with a p-time winning strategy, since by Theorem 4.5 of [24], there is a recursive game with unique winning strategy, which is recursive but not p-time.

Corollary 12.2 *For each recursive instance P of any of the problems listed in Theorem 12.1, there is a p-time instance Q of the problem such that the Π_1^0 class of solutions to P is recursively homeomorphic to the Π_1^0 class of solutions to Q .*

Proof. In each case, it is easy to see that the recursive isomorphism between P and Q gives rise to a recursive homeomorphism between the Π_1^0 classes of solutions.

For example, we consider the coloring problem. Recall that the Π_1^0 class of k -colorings on a recursive graph $G = (V, E)$ (where V may be assumed to equal ω) is the set $[T]$ of infinite paths through the recursive k -ary tree T , where a finite sequence $(\sigma(0), \dots, \sigma(n-1)) \in \{1, 2, \dots, k\}^n$ is in T if and only if $\sigma(i) \neq \sigma(j)$ for all $(i, j) \in E$. Now suppose that f is a recursive isomorphism mapping G to the recursive graph $G' = (V', E')$, so that $V' = \{f(0), f(1), \dots\}$ and $(f(i), f(j)) \in E'$ if and only if $(i, j) \in E$. Then we can define the tree $(k+1)$ -ary T' by having $(\tau(0), \dots, \tau(n-1)) \in \{0, 1, \dots, k\}^n$ in T' if and only if

- (1) $\tau(v) = 0 \iff v \notin V'$;
- (2) $\tau(u) \neq \tau(v)$ whenever $(u, v) \in E'$.

Then $[T']$ represents in a reasonable way the set of legal k -colorings on G' and we have a natural homeomorphism from $[T]$ to $[T']$ defined by $H(x)(f(i)) = x(i)$ and $H(x)(v) = 0$ if $v \notin V'$. $\square$

We can now represent Π_1^0 classes as the set of solutions to p-time problems of the types listed above. We list only some of the results.

Corollary 12.3 *For each of the problems (1) through (9), and (13) listed in Theorem 12.1,*

- (a) *The problem of finding a recursive solution to a p-time problem can strongly represent the r.b. Π_1^0 class of separating sets for any pair of disjoint infinite r.e. sets.*

- (b) *There is a p-time instance of the problem with no recursive solution.*
- (c) *If $\mathbf{a}$ is a Turing degree and $\mathbf{0} <_T \mathbf{a} <_T \mathbf{0}'$, then there is a p-time instance P of the problem such that P has a solution of degree $\mathbf{a}$ but has no recursive solution.*

For problems (1), (3), (6) and (13), we have also:

- (d) *The problem of finding a recursive solution to a p-time problem can strongly represent an arbitrary r.b. Π_1^0 class.*
- (e) *There exists a p-time instance P of the problem such that*
 - (i) *P has a unique non-recursive solution y which is also the unique limit solution and has degree $\mathbf{0}'$ and such that any other solution is recursive,*
 - (ii) *if R is any recursive sub-problem of P and z is any recursive solution of R , then either (I) there are only finitely many solutions of P which extend z , or (II) all but finitely many solutions of P extend z .*
 - (iii) *if x is any recursive solution of P , then there is some finite sub-problem F of P such that any solution of P which agrees with x on F must equal x .*

We have seen that, by changing the names of the vertices, we can transform a recursive graph into a p-time graph. However, we would prefer for a countably infinite graph to have the set V of vertices equal to some standard universe such that the tally or binary representation of the set of natural numbers. This would, for instance, allow us to define the homeomorphism of Corollary 12.2 without worrying about the set of non-vertices. The p-time graph constructed by Theorem 11.2 will have a rather sparse set of vertices and this appears to be an essential part of the theorem.

We will next indicate how to fill out the p-time structure given by Theorem 12.1 to a structure with universe $\text{Bin}(\omega)$ such that there is a degree preserving correspondence, which is one-to-one (up to a finite permutation), between the Π_1^0 classes of solutions of the associated problems. For example, in the coloring problem, we add vertices whose colors will be determined, up to a permutation, by the coloring of the vertices of Q .

Theorem 12.4 *For each recursive instance P of the combinatorial problems listed below, there is a p-time instance Q with universe $\text{Bin}(\omega)$ and a degree preserving correspondence between the solutions of P and the solutions of Q .*

- (1) *Finding a k -coloring for a k -colorable highly recursive graph, for any $k \geq 3$.*
- (2) *Finding a marriage in a highly recursive society.*
- (3) *Finding a surjective marriage in a symmetrically highly recursive society.*
- (4) *Finding a surjective marriage in a symmetrically highly recursive society where each person knows at most two other people.*
- (5) *Finding a k -partition of a highly recursive graph such that no set in the partition is adjacent to m other sets.*
- (6) *Finding a (one-way or two-way) Hamiltonian or Euler path for a highly recursive graph.*
- (7) *Covering a recursive poset of width k by k chains, for any $k \geq 2$.*
- (8) *Covering a recursive poset of height k by k antichains, for any $k \geq 2$.*
- (9) *Expressing a recursive partial ordering on a set as the intersection of d linear orderings on the set.*
- (10) *Finding an ω -successivity (or an ω^* -successivity) in a recursive linear ordering.*

Proof. In each case, we may assume by Corollary 12.2 that we start with a p-time instance of the problem which is a relational structure $\mathcal{B}$ with some universe $B \subseteq \text{Bin}(\omega)$. Now it follows from Lemma 2.3 of [23] that $\mathcal{B}$ is recursively isomorphic to a p-time structure $\mathcal{A}$ with universe $A \subseteq \text{Tal}(\omega)$. Then Lemma 2.6 of [23] says that the disjoint union $A \oplus \text{Bin}(\omega)$ is p-time isomorphic to $\text{Bin}(\omega)$, where

$$X \oplus Y = \{\langle 0, x \rangle : x \in X\} \cup \{\langle 1, y \rangle : y \in Y\}.$$

Then we will create a p-time structure $\mathcal{C}$ with universe $A \oplus \text{Bin}(\omega)$ which has a copy of $\mathcal{A}$ together with a copy of $\text{Bin}(\omega)$, where the relations will

be defined on $\text{Bin}(\omega)$ and between A and $\text{Bin}(\omega)$ so as to determine the degree preserving correspondence between the solutions of $\mathcal{A}$ and those of the extension $\mathcal{C}$. Since the universe C of $\mathcal{C}$ is p-time isomorphic to $\text{Bin}(\omega)$, it follows from Lemma 2.2 of [23] that $\mathcal{C}$ is recursively isomorphic to a p-time structure with universe $\text{Bin}(\omega)$. Then we will let Q be the problem associated with this structure. It follows that there will be a degree preserving correspondence between the set of solutions of Q and the set of solutions of the original problem P . In each case, we will assume that our original structure is p-time and has for its universe a p-time subset A of $\text{Tal}(\omega)$ and that there is a p-time list of $\text{Bin}(\omega) \setminus A$. These assumptions are justified by the above discussion. In each case, the correspondence will be one-to-one unless otherwise indicated.

(1) Finding a k -coloring for a k -colorable highly recursive graph, for any $k \geq 3$.

This is Theorem 2.1 of [22]. Here the correspondence is one-to-one, up to a finite permutation of the colors on the new vertices.

(2) Finding a marriage in a highly recursive society.

Let $S = (B, G, K)$ be a p-time society. Then we will directly extend S to a highly recursive p-time society $S' = (B', G', K')$ where $B' = G' = \text{Bin}(\omega)$. Let $\text{Bin}(\omega) \setminus B = \{b_0, b_1, \dots\}$ and let $\text{Bin}(\omega) \setminus G = \{g_0, g_1, \dots\}$ be p-time lists of the new boys and girls in the society S' . Then K' is defined by putting $(b_i, g_i) \in K'$ for all i . It is clear that any marriage f on S has a unique extension f' to S' defined by letting $f'(b_i) = g_i$ for all i . It follows that f and f' have the same degree.

(3) Finding a surjective marriage in a symmetrically highly recursive society.

The extension is the same as in (2). It is clear that f' will be onto if and only if f is onto.

(4) Finding a surjective marriage in a symmetrically highly recursive society where each person knows at most two other people.

The extension is again the same as in (2). It is clear that if each person in S knows at most two other people, then each person in the extension S' also knows at most two other people.

(5) Finding a k -partition of a highly recursive graph such that no set in the partition is adjacent to m other sets, with $m > 2$.

Let the p-time graph $G = (V, E)$ be given. We define a p-time graph $G_1 = (V_1, E_1)$ to be a regular $(m-1)$ -ary tree of complete k -graphs. That is, define the regular $(m-1)$ -ary tree T_{m-1} to consist of a root node $\emptyset$ together with the set

$$\{\text{bin}(0), \text{bin}(1), \dots, \text{bin}(m-1)\} \times \{\text{bin}(1), \text{bin}(2), \dots, \text{bin}(m-2)\}^*,$$

where $\emptyset$ has $m-1$ successors $(\text{bin}(i), \emptyset)$ for $i < m$ and $(\text{bin}(i), \sigma)$ has $m-2$ successors $(\text{bin}(i), \sigma \frown \text{bin}(j))$ for $j < m-1$. Then we let

$$V_1 = \{\text{bin}(1), \text{bin}(2), \dots, \text{bin}(k)\} \times T_{m-1},$$

and we let $((\text{bin}(i), \sigma), (\text{bin}(j), \tau)) \in E_1$, where $\sigma = (\sigma(1), \dots, \sigma(m-1))$ and $\tau = (\tau(1), \dots, \tau(m-1))$, provided that $\sigma = \tau$ or either τ is a successor of σ or σ is a successor of τ . It is clear that if the graph is recursively partitioned into the complete k -graphs corresponding to the nodes of T_{m-1} , then each set in the partition is adjacent to at most $m-1$ other sets. We see also that each node of T_{m-1} has $m-1$ neighbors, so that any two distinct nodes have at least $2m-4$ other neighbors. Now let $\{A_i : i < \omega\}$ be a k -partition of G_1 . Suppose that some A_i contains vertices u and v corresponding to different nodes of the T_{m-1} . Then u and v taken together have at least $2(k-1) + (2m-4)k = (2m-2)k - 2$ other adjacent vertices in G_1 . Since $k-2$ of these could belong to A_i , we see that A_i has at least $(2m-3)k$ adjacent vertices. Since each set in the partition has at most k vertices, it follows that A_i is adjacent to at least $2m-3$ sets in the partition. Thus since $m > 2$, we have $2m-3 > m-1$ so that the set A_i is adjacent to too many sets.

Now let $G' = G \oplus G_1$. It is clear that for any k -partition of G , there is a partition of G' of the same degree which is given by adding the recursive partition of G_1 into the nodes of the tree as defined above. We claim that these are the only possible partitions. That is, suppose $\{B_j : j < \omega\}$ is a k -partition of G' . It suffices to show that for any $u \in V_1$ and any j , if $u \in B_j$, then the entire node to which u belongs must be included in B_j . Suppose that this is false. It follows from the argument above that B_j may not contain an element of a different node of T . Thus the set B_j has all $k(m-1)$ vertices from the adjacent nodes as neighbors as well as at least one vertex from the node of u . But this clearly implies that at least m sets of the partition must be adjacent to B_j .

(6) Finding a one-way (or two-way) Hamiltonian Euler path starting from a fixed vertex for a highly recursive graph.

Let the p-time graph $G = (V, E)$ be given with $V = \{v_0 < v_1 < \dots\}$ a subset of $Tal(\omega)$. Let each edge $(tal(m), tal(n))$ of V with $tal(m) < tal(n)$ be coded as $0^{n+1}1^{m+1}$ in $Bin(\omega)$. Let $b_0, b_1, \dots$ enumerate the codes of edges in increasing order and let $b_i = 0^{n_i+1}1^{m_i+1}$ for each i .

Now let $V' = V \oplus Bin(\omega)$ and let E' be defined by joining $\langle 1, b_i \rangle$ to $\langle 0, tal(m_i) \rangle$, joining $\langle 0, tal(n_i) \rangle$ to $\langle 1, b_{i+1} - 1 \rangle$, joining $\langle 1, b \rangle$ to $\langle 1, b + 1 \rangle$ whenever $b + 1 \neq b_i$ for any i , and joining $\langle 1, b_0 - 1 \rangle$ to $\langle 0, tal(m_0) \rangle$. Note that other than the initial sequence of edges connecting $\langle 1, 0 \rangle$ to $\langle 1, b_0 - 1 \rangle$ and then to $\langle 0, tal(m_0) \rangle$, this has the effect of replacing an edge $(m, n) \in E$ where $m < n$ and $b_i = 1^{m+1}0^{n+1}$ by a sequence of edges

$$(\langle 0, m \rangle, \langle 1, b_i \rangle), (\langle 1, b_i \rangle, \langle 1, b_i + 1 \rangle), \dots, \\ (\langle 1, b_{i+1} - 2 \rangle, \langle 1, b_{i+1} - 1 \rangle), (\langle 1, b_{i+1} - 1 \rangle, \langle 0, n \rangle).$$

See Figure 6.

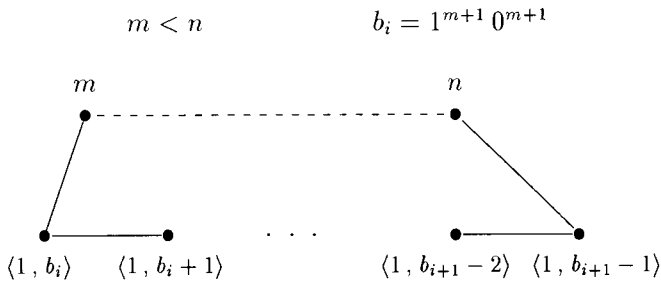


Figure 6: Replacement for edge (m, n)

Thus to test whether $\langle 1, b \rangle$ and $\langle 1, c \rangle$ are joined by an edge, where $b < c$, we simply check that $c = b + 1$ and that, if $c = 0^{n+1}1^{m+1}$ with $m < n$, then $(tal(m), tal(n)) \notin E$. To determine whether $\langle 0, v \rangle$ and $\langle 1, c \rangle$ are joined by an edge, we first check that $v = tal(m) \in V$ and that either

(i) $c = 0^{s+1}1^{r+1}$, or

(ii) $c = 0^{s+1}1^{r+1} - 1$, for some edge $(tal(r), tal(s))$ in G with $r < s$.

Finally, in case (i), we check that $r = m$ and, in case (ii), we either check that $m = m_0$ and that $c + 1 = b_0$ or else we compute the largest code $0^{q+1}1^{p+1}$ of an edge of G less than $c + 1$ and check that $m = q$. If everything checks, then there is an edge and otherwise there is not. Thus G' is a p-time graph.

Now suppose that f is a one-way Euler path on G starting from $f(0) = v_0 = \text{tal}(m_0)$. Then we can define a corresponding Euler path on G' starting from $\langle 1, \text{bin}(0) \rangle$ by beginning with the sequence

$$\langle 1, \text{bin}(0) \rangle, \langle 1, \text{bin}(1) \rangle, \dots, \langle 1, b_0 - 1 \rangle, \langle 0, v_0 \rangle$$

and then replacing in turn each edge $(f(i), f(i+1))$ which joins $\text{tal}(m)$ to $\text{tal}(n)$ with $m < n$, either by the sequence

$$\langle 0, f(i) \rangle, \langle 1, \text{bin}(b) \rangle, \langle 1, \text{bin}(b+1) \rangle, \dots, \langle 1, \text{bin}(c-1) \rangle, \langle 0, f(i+1) \rangle$$

if $f(i) = \text{tal}(m) < f(i+1)$, or by the sequence

$$\langle 0, f(i) \rangle, \langle 1, \text{bin}(c-1) \rangle, \langle 1, \text{bin}(c-2) \rangle, \dots, \\ \langle 1, \text{bin}(b+1) \rangle, \langle 1, \text{bin}(b) \rangle, \langle 0, f(i+1) \rangle$$

if $f(i) = \text{tal}(n) > f(i+1)$, where $b = 0^{n+1}1^{m+1}$ and c is the least code greater than b for an edge of G . It is clear that this one-way Euler path is recursive in f .

Conversely, let g be a one-way Euler path in G' starting from $\langle 1, \text{bin}(0) \rangle$. It is clear that the path must proceed through

$$\langle 1, \text{bin}(1) \rangle, \langle 1, \text{bin}(2) \rangle, \dots, \langle 1, b_0 - 1 \rangle$$

and then to $\langle 0, v_0 \rangle$. Now let $f: \omega \rightarrow V$ be defined by letting $\langle 0, f(i) \rangle$ be the i -th vertex of the form $\langle 0, x \rangle$ in the path g . Then f is a one-way Euler path for G and it follows from the construction that g is the corresponding path as defined above, since there is only one way in G' to go from $\langle 0, f(i) \rangle$ to $\langle 0, f(i+1) \rangle$.

For two-way Euler paths, modify the construction by eliminating the finite initial sequence

$$\langle 1, \text{bin}(0) \rangle, \langle 1, \text{bin}(1) \rangle, \dots, \langle 1, \text{bin}(m_0) - 1 \rangle$$

of vertices of G' along with the edges through those vertices. Then the remaining vertex set is still p-time isomorphic to $\text{Bin}(\omega)$ and the argument goes through as above.

The Hamiltonian paths require a different construction. We assume without loss of generality that the vertices of G include 0 and that all are multiples of 4 (in binary) and let $4m_0, 4m_1, \dots$ enumerate the vertices of G in increasing order. Define the graph G' to have vertex set $V' = \text{Bin}(\omega)$ with edges defined as follows. For each i , there will be two sequences of edges joining the set of binary numbers from $4m_i + 1$ up to $4m_{i+1}$, as follows.

- (i) $4m_{i+1}, 4m_{i+1}-4, 4m_{i+1}-8, \dots, 4m_i+4, 4m_i+2, 4m_i+6, \dots, 4m_{i+1}-2.$
- (ii) $4m_{i+1}, 4m_{i+1}-3, 4m_{i+1}-7, \dots, 4m_i+1, 4m_i+3, \dots, 4m_{i+1}-1.$

These are the vertices associated with $4m_{i+1}$. In addition, for each edge joining $4m_i$ and $4m_j$ in G with $m_i, m_j \neq 0$, there are edges joining $4m_i - 1$ with $4m_j - 2$ and joining $4m_i - 2$ with $4m_j - 1$. For an edge in G joining $4m_i$ with 0, there is an edge joining $4m_i - 1$ with 0. The procedure for determining whether there is an edge joining a and b is the following. First look for the largest m and n such that $4m \in V$ and $4m < a$ and $4n \in V$ and $4n < b$. In the special case that $a = 0$, a and b are joined if and only if $b + 1$ is joined to 0 as vertices in G . Otherwise there are several cases. First suppose that $m = n$; then a and b are joined if and only if, either they differ by exactly 4 or $\{a, b\} = \{4m + 1, 4m + 3\}$ or $\{a, b\} = \{4m + 2, 4m + 4\}$. Next suppose that $m \neq n$. Then a and b are joined if and only if either $a + 1$ and $b + 2$ are joined as vertices in G or $a + 2$ and $b + 1$ are joined as vertices in G . Thus G' is a p-time graph.

Now let f be a one-way Hamiltonian path on G starting from $v_0 = 0$ and suppose that $f(i) = 4m_{r_i}$. Then there is a corresponding Hamiltonian path g in G' obtained by replacing the edge from v_0 to $4m_{r_1}$ with the sequence of edges joining v_0 to $4m_{r_1} - 1$ and then on to $4m_{r_1} - 3$ and $4m_{r_1}$ as described above, and for $i > 0$, replacing each edge $(f(i), f(i + 1))$ with the sequence of edges first joining $4m_{r_i}$ to $4m_{r_i} - 4$ and then on to $4m_{r_i} - 2$ as described above, then joining $4m_{r_i} - 2$ to $4m_{r_{i+1}} - 1$, and closing with the sequence joining $4m_{i+1} - 1$ to $4m_{i+1} - 3$ and then $4m_{i+1}$. Thus for each $i > 0$, the even vertices associated with $f(i)$ are joined to the odd vertices associated with $f(i + 1)$.

Conversely, let g be a one-way Hamiltonian path in G' starting from $v_0 = 0$ and define $f(i) = 4m_{r_i}$ so that $0, 4m_{r_1}, \dots$ lists the members of G in order of appearance in the path g . It follows from the construction that f is a one-way Hamiltonian path for G starting from v_0 and that g is the corresponding path as defined above.

For the two-way Hamiltonian paths, the construction is modified by adding an edge joining v_0 with $4m_i - 2$ for each edge joining v_0 with $4m_i$ in G . Then for any two way Hamiltonian path f in G , there will be two corresponding two-way paths in G' , one in which the even vertices associated with $f(i)$ are joined to the odd vertices associated with $f(i+1)$ and one in which the odd vertices associated with $f(i)$ are joined with the even vertices of $f(i+1)$. Thus the correspondence here is two-to-one.

(7) The problem of covering a recursive poset of width k by k chains, for any $k \geq 2$.

Let $\mathcal{P} = (P, \leq_P)$ be a p-time poset where $P \subseteq \text{Tal}(\omega)$. Then define a p-time poset $\mathcal{R} = (R, \leq_R)$ where

$$R = P \oplus (\{ \text{bin}(1), \dots, \text{bin}(k) \} \times \text{Bin}(\omega))$$

and $\langle 0, p \rangle \leq_R \langle 0, q \rangle$ iff $p \leq_P q$, $\langle 0, p \rangle \leq_R \langle 1, n \rangle$ for all p and n , and $\langle 1, m \rangle \leq_R \langle 1, n \rangle$ iff $m = \langle \text{bin}(i), \text{bin}(r) \rangle$ and $n = \langle \text{bin}(i), \text{bin}(s) \rangle$ where $r \leq s$.

Then it is clear that any covering f of $\mathcal{P}$ by k chains induces a covering f' of covering of $\mathcal{R}$ by k chains where

- (i) $f'(\langle 0, p \rangle) = f(p)$ for all $p \in P$ and
- (ii) $f'(\langle 1, \langle \text{bin}(i), n \rangle \rangle) = f'(\langle 1, \langle \text{bin}(i), m \rangle \rangle)$ for all i, m and n .

Thus the covering is determined by the value of f' on the finitely many new points

$$\langle 1, \langle \text{bin}(1), 0 \rangle \rangle, \dots, \langle 1, \langle \text{bin}(1), 0 \rangle \rangle.$$

This shows that f' has the same degree as f and that f' is unique up to a permutation of the names of chains. Then $\mathcal{R}$ is p-time isomorphic to p-time linear ordering $\mathcal{S}$ whose universe is $\text{Bin}(\omega)$.

(8) The problem of covering a recursive poset of height k by k antichains, for any $k \geq 2$.

This is the dual of problem (7). The partial order is now defined by making

$$\langle 1, \langle \text{bin}(i), n \rangle \rangle \leq \langle 1, \langle \text{bin}(j), m \rangle \rangle \iff (i < j \ \& \ n = m).$$

(9) The dimension of posets problem.

Let $\mathcal{P} = (\mathcal{P}, \leq_{\mathcal{P}})$ be a poset and let $\text{Bin}(\omega) \setminus P = \{v_i : i < \omega\}$. The partial order $\leq'$ is defined on $\text{Bin}(\omega)$ by making $p \leq' v_i$ for all $p \in P$ and all i and making $v_i \leq' v_j$ if and only if $v_i \leq v_j$ (where $<$ is the usual ordering on $\text{Bin}(\omega)$).

(10) Finding an ω -successivity (or an ω^* -successivity) in a recursive linear ordering.

Given a p-time linear ordering $L_1 = (A, <_1)$ on a p-time set

$$A = \{a_0 < a_1 < \cdots\},$$

we may assume $a_0 = 0$ and that each $a_i = \text{bin}(4m_i)$. Now define the p-time ordering $L_2 = (\text{Bin}(\omega), <_2)$ by replacing each point $a = \text{bin}(4m_i)$ with a block $B(a)$:

$$\begin{aligned} \text{bin}(4m_i + 1) &< \text{bin}(4m_i + 5) < \cdots < \text{bin}(4m_{i+1} - 3) \\ &< \text{bin}(4m_{i+1} - 1) < \text{bin}(4m_{i+1} - 5) < \cdots < \text{bin}(4m_i + 3) \\ &< \text{bin}(4m_i) < \text{bin}(4m_i + 4) < \text{bin}(4m_i + 8) < \cdots < \text{bin}(4m_{i+1} - 4) \\ &< \text{bin}(4m_{i+1} - 2) < \text{bin}(4m_{i+1} - 6) < \cdots < \text{bin}(4m_i + 2) \end{aligned}$$

That is, we use the elements between $4m_i$ and $4m_{i+1}$ which are equivalent to 1 mod 4 to form a chain between $4m_i + 1$ and $4m_{i+1} - 1$, then we use the elements between $4m_i$ and $4m_{i+1}$ which are equivalent to 3 mod 4 in reverse order to form a chain between $4m_{i+1} - 1$ and $4m_i$, etc..

Now suppose that f is an ω -successivity in L_1 . Then we can recursively obtain an ω -successivity g in L_2 by replacing each point $f(i)$ with the block $B(f(i))$. Conversely, given an ω -successivity g in L_2 , the ω -successivity f of L_1 may be defined by making $f(i)$ the i -th binary number in the successivity g which is divisible by 4 and it then follows that g is the successivity obtained from f as above. The argument for ω^* -successivities is similar. $\square$

We remark that, for the 3-coloring problem, it is possible to improve this result by having the 3-colorings of the the original recursive graph be restrictions of the 3-colorings of the p-time graph to the original recursive vertex set.

Theorem 12.4 can now be applied to obtain improved versions of Corollary 12.3. We list only a few here.

Corollary 12.5

- (a) *There exists a p -time graph G with universe $\text{Bin}(\omega)$ which has a unique non-recursive Hamiltonian path π , where π has degree $\mathbf{0}'$ and such that any other Hamiltonian path is the unique extension to G of a Hamiltonian path on some finite subgraph F of G .*
- (b) *There is a p -time partial ordering with universe $\text{Bin}(\omega)$ of width k which has no recursive covering by k chains.*
- (c) *For any $x \leq_T \mathbf{0}'$, there is a p -time linear ordering $\mathcal{A}$ with universe $\text{Bin}(\omega)$ such that there is ω -successivity (respectively ω^* -successivity) of $\mathcal{A}$ of degree x and every ω -successivity (respectively ω^* -successivity) of $\mathcal{A}$ is either recursive or has the same Turing degree as x .*

Theorem 12.4 and Corollary 12.5 demonstrate that the problem of finding solutions to feasible problems is just as difficult as the problem of finding solutions to recursive problems. Therefore more conditions will have to be put on a problem than just feasibility if our goal is to guarantee the existence of a feasible solution, or even the existence of a recursive solution. There are many possible approaches to this goal, some of which were explored in [22] for the graph-coloring problem.

Finally, we consider the problem of finding a prime ideal of a recursive Boolean algebra, or more generally, of a recursive ring.

Theorem 12.6 *For any recursive Boolean algebra $\mathcal{B}$, there is a p -time commutative ring $\mathcal{R}$ with unity, having universe $\text{Bin}(\omega)$, and a one-to-one degree preserving map between the class of prime ideals of $\mathcal{R}$ and the class of prime ideals of $\mathcal{B}$.*

Proof. By Theorem 12.1, we may assume that $\mathcal{B}$ is a p -time Boolean algebra, and thus a Boolean ring. Now define the ring $\mathcal{R} = \mathcal{B} \oplus \mathbb{Q}$. $\mathbb{Q}$ is chosen here because it has no (proper) prime ideals. The ring $\mathbb{Q}$ of rationals may be represented as a p -time ring with universe $\text{Bin}(\omega)$, and it follows from Lemmas 2.2 and 2.6 of [23] that $\mathcal{R}$ is p -time isomorphic to a ring with universe $\text{Bin}(\omega)$. For any prime ideal I of $\mathcal{B}$, it is easy to check that $I \oplus \mathbb{Q}$ is a prime ideal of $\mathcal{R}$ and that these are the only prime ideals of $\mathcal{R}$. $\square$

Corollary 12.7

- (i) *For any degree $\mathbf{a} <_T \mathbf{0}'$, there exists a recursive commutative ring $\mathcal{R}$ with a prime ideal I of degree $\mathbf{a}$ such that I is the unique non-recursive prime ideal of $\mathcal{B}$ and such that any other prime ideal of $\mathcal{B}$ is finitely generated.*
- (ii) *There is a recursive commutative ring with unity, $\mathcal{R}$, which has a unique non-recursive prime ideal I , such that any other prime ideal of $\mathcal{R}$ is finitely generated, and such that for any r.e. ideal J of $\mathcal{R}$, either there are only finitely many prime ideals of $\mathcal{R}$ extending J or else all but finitely many of the prime ideals of $\mathcal{R}$ extend J .*

References

- [1] R. Aharoni, M. Magidor and R. A. Shore, On the strength of König's duality theorem for infinite bipartite graphs, *J. Combin. Theory, Ser. B*, **54** (1992) 257–290.
- [2] K. I. Appel and W. Haken, Every planar map is four colorable. Part I: Discharging, *Illinois J. Math.*, **21** (1977) 429–490.
- [3] K. I. Appel, W. Haken and J. Koch, Every planar map is four colorable. Part II: Reducibility, *Illinois J. Math.*, **21** (1977) 491–567.
- [4] K. I. Appel, W. Haken and J. Koch, Every planar map is four colorable, *Contemp. Math.*, **98** (1989).
- [5] K. R. Apt, Logic programming, in: *Handbook of Theoretical Computer Science*, vol. B, Formal Models and Semantics, J. van Leeuwen (ed.), (Elsevier, Amsterdam, MIT Press, Cambridge, Mass, 1990) 493–574.
- [6] E. Artin and O. Schreier, Algebraische Konstruktion reeller Körper, *Abh. Math. Sem. Univ. Hamburg*, **5** (1927) 85–99.
- [7] D. R. Bean, Effective coloration, *J. Symbolic Logic*, **41** (1976) 469–480.
- [8] D. R. Bean, Recursive Euler and Hamilton paths, *Proc. Amer. Math. Soc.*, **55** (1976) 385–394.

- [9] R. Beigel and W. I. Gasarch, On the complexity of finding the chromatic number of a recursive graph I: The bounded case, *Ann. Pure Appl. Logic*, **45** (1989) 1–38.
- [10] R. Beigel and W. I. Gasarch, On the complexity of finding the chromatic number of a recursive graph II: The unbounded case, *Ann. Pure Appl. Logic*, **45** (1989) 227–246.
- [11] L. Blum, M. Shub and S. Smale, On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines, *Bull. Amer. Math. Soc. (N.S.)*, **21** (1989) 1–46.
- [12] R. L. Brooks, On colouring the nodes of a network, *Proc. Cambridge Philos. Soc.*, **37** (1941) 194–197, [also in: *Classic Papers in Combinatorics*, I. Gessel and G.-C. Rota (eds.), (Birkhäuser, Boston, 1987), 118–121].
- [13] H.-G. Carstens, A 5-color extension theorem, in: *Computation Theory and Logic*, E. Börger, (ed.), *Lecture Notes in Comput. Sci.*, **270** (1987) 67–77.
- [14] D. Cenzer, Effective real dynamics, in: *Logical Methods*, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), *Progr. Comput. Sci. Appl. Logic*, **12** (1993) 162–177.
- [15] D. Cenzer, Π_1^0 classes, in: *Handbook of Recursion Theory*, E. R. Griffor, (ed.), North-Holland, (to appear).
- [16] D. Cenzer, P. Clote, R. L. Smith, R. I. Soare and S. S. Wainer, Members of countable Π_1^0 classes, (Special issue: 2nd. Southeast Asian Logic Conf., Bangkok, 1984) *Ann. Pure Appl. Logic*, **31** (1986) 145–163.
- [17] D. Cenzer, R. G. Downey, C. G. Jockusch and R. A. Shore, Countable thin Π_1^0 classes, *Ann. Pure Appl. Logic*, **59** (1993) 79–139.
- [18] D. Cenzer and E. Howorka, On vertex k -partitions of certain infinite graphs, *Discrete Math.*, **23** (1978) 105–113.
- [19] D. Cenzer, W. Marek, and J. B. Remmel, Index sets for finite predicate logic programs, (to appear).

- [20] D. Cenzer and R. D. Mauldin, On the Borel class of the derived set operator, *Bull. Soc. Math. France*, **110** (1982) 357–380, and II, **111** (1983) 367–372.
- [21] D. Cenzer and J. B. Remmel, Polynomial-time versus recursive models, *Ann. Pure Appl. Logic*, **54** (1991) 17–58.
- [22] D. Cenzer and J. B. Remmel, Feasible graphs and colorings, *Math. Logic Quart.*, **41** (1992) 327–352.
- [23] D. Cenzer and J. B. Remmel, Polynomial-time abelian groups, *Ann. Pure Appl. Logic*, **56** (1992) 313–363.
- [24] D. Cenzer and J. B. Remmel, Recursively presented games and strategies, *Math. Social Sciences*, **24** (1992) 117–139.
- [25] D. Cenzer and J. B. Remmel, Complexity theoretic model theory and algebra, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 381–513.
- [26] D. Cenzer and J. B. Remmel, Index sets for Π_1^0 classes, *Ann. Pure Appl. Logic*, (to appear).
- [27] D. Cenzer and J. B. Remmel, Index sets in computable analysis, (to appear).
- [28] D. Cenzer and R. L. Smith, On the ranked points of a Π_1^0 set, *J. Symbolic Logic*, **54** (1989) 975–991.
- [29] P. A. Cholak, R. Coles and R. G. Downey, Invariance, degrees and Π_1^0 classes, (to appear).
- [30] P. A. Cholak and R. G. Downey, On the Cantor-Bendixson rank of recursively enumerable sets, *J. Symbolic Logic*, **58** (1993) 629–640.
- [31] P. Clote, On recursive trees with a unique infinite branch, *Proc. Amer. Math. Soc.*, **93** (1985) 335–342.
- [32] T. Craven, The Boolean space of orderings of a field, *Trans. Amer. Math. Soc.*, **209** (1975) 225–235.

- [33] J. C. E. Dekker and J. Myhill, Retractable sets, *Canad. J. Math.*, **10** (1958) 357–373.
- [34] R. P. Dilworth, A decomposition theorem for partially ordered sets, *Ann. of Math.*, Ser. 2, **51** (1950) 161–166.
- [35] R. G. Downey, Abstract Dependence, Recursion Theory and the Lattice of Recursively Enumerable Filters, Ph.D. Thesis, Monash Univ., Clayton, Victoria, Australia, (1982).
- [36] R. G. Downey, Maximal theories, *Ann. Pure Appl. Logic*, **33** (1987) 245–282.
- [37] R. G. Downey, On Π_1^0 classes and their ranked points, *Notre Dame J. Formal Logic*, **32** (1991) 499–512.
- [38] R. G. Downey, On presentations of algebraic structures, in: Complexity, Logic and Recursion Theory, A. Sorbi, (ed.), *Lecture Notes in Pure and Appl. Math.*, **187** (1997) 157–205.
- [39] R. G. Downey, Computability theory and linear orderings, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 823–976.
- [40] R. G. Downey and C. G. Jockusch Jr., Effective presentability of Boolean algebras of Cantor–Bendixson rank 1, *J. Symbolic Logic*, (to appear).
- [41] R. G. Downey, C. G. Jockusch Jr. and M. Stob, Array nonrecursive sets and multiple permitting arguments, in: *Recursion Theory Week*, (Proc. Conf., Oberwolfach, Mar. 19–25, 1989), K. Ambos-Spies, G. H. Müller and G. E. Sachs, (eds.), *Lecture Notes in Math.*, **1432** (1990) 141–173.
- [42] R. G. Downey and S. A. Kurtz, Recursion theory and ordered groups, *Ann. Pure Appl. Logic*, **32** (1986) 137–151.
- [43] R. G. Downey and S. Lempp, On the proof theoretical strength of the Dushnik-Miller theorem for countable linear orderings, (in preparation).
- [44] R. G. Downey and M. Moses, Recursive linear orders with incomplete successivities, *Trans. Amer. Math. Soc.*, **326** (1991) 653–668.

- [45] R. G. Downey and Y. Yang, A rank one cohesive set, *Ann. Pure Appl. Logic*, **68** (1994) 161–171.
- [46] B. Dushnik and E. W. Miller, Partially ordered sets, *Amer. J. Math.*, **63** (1941) 600–610.
- [47] A. Ehrenfeucht, Separable theories, *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, **9** (1961) 17–19.
- [48] L. J. Feiner, Orderings and Boolean Algebras not Isomorphic to Recursive Ones, Ph.D. Thesis, MIT, Cambridge, MA, (1967).
- [49] L. J. Feiner, Hierarchies of Boolean algebras, *J. Symbolic Logic*, **35** (1970) 365–374.
- [50] A. P. Ferry, Enriched Nonmonotonic Rule Systems, M.Sc. Thesis, Univ. Kentucky, Lexington, KY, (1991).
- [51] A. P. Ferry, A topological characterization of stable and minimal model classes of propositional logic programs, *Ann. Math. Artificial Intelligence*, **15** (1995) 325–355.
- [52] A. P. Ferry, Characterizations and properties of classes associated with logic programming semantics, (to appear).
- [53] K. Ko and H. M. Friedman, Computational complexity of real functions, *Theoret. Comput. Sci.*, **20** (1982) 323–352.
- [54] H. M. Friedman, S. G. Simpson and R. L. Smith, Countable algebra and set existence axioms, *Ann. Pure Appl. Logic*, **25** (1983) 141–181; addendum, *ibid.* **27** (1985) 319–320.
- [55] A. Fröhlich and J. C. Shepherdson, Effective procedures in field theory, *Philos. Trans. Roy. Soc. London, Ser. A*, **248** (1956) 407–432.
- [56] D. Gale and F. M. Stewart, Infinite games with perfect information, in: *Contrib. Theory of Games*, Vol. 2, H. W. Kuhn and A. W. Tucker, (eds.), *Annals of Math. Stud.*, **28** (1953) 245–266.
- [57] W. I. Gasarch, A survey of recursive combinatorics, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 1041–1176.

- [58] W. I. Gasarch and G. Martin, Index sets in recursive combinatorics, in: *Logical Methods*, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), *Progr. Comput. Sci. Appl. Logic*, **12** (1993) 352–385.
- [59] M. Gelfond and V. Lifschitz, The stable model semantics for logic programming, in: *Logic Programming*, (Proc. 5th. Internatl. Sympos. Logic Programming, Univ. Washington, Seattle, Wash., Aug. 15–19, 1988), R. A. Kowalski and K. A. Bowen, (eds.), (MIT Press, Cambridge, Mass., 1988), 1070–1080.
- [60] M. Gelfond and V. Lifschitz, Logic programs with classical negation, in: *Logic Programming*, (Proc. 7th. Internatl. Conf. Logic Programming, Jerusalem, June 18–20, 1990), D. H. D. Warren and P. Szeredi (eds.), (MIT Press, Cambridge, Mass., 1990) 579–597.
- [61] S. Grigorieff, Every recursive linear ordering has a copy in $\text{DTIME-SPACE}(n, \log(n))$, *J. Symbolic Logic*, **55** (1990) 260–276.
- [62] M. J. Groszek and T. A. Slaman, Π_1^0 classes and minimal degrees, (Special issue: *Logic Colloq. '95*, Haifa), *Ann. Pure Appl. Logic*, **87** (1997) 117–144.
- [63] M. Hall, Distinct representatives of subsets, *Bull. Amer. Math. Soc.*, **54** (1948) 922–926.
- [64] D. Harel, Hamiltonian paths in infinite graphs, *Israel J. Math.*, **76** (1991) 317–336.
- [65] L. S. Hay, A. B. Manaster and J. G. Rosenstein, Concerning partial recursive similarity transformations of linearly ordered sets, *Pacific J. Math.*, **71** (1977) 57–70.
- [66] P. G. Hinman, Recursion-Theoretic Hierarchies, *Perspect. in Math. Logic*, (1978).
- [67] J. L. Hirst, Marriage theorems and reverse mathematics, in: *Logic and Computation*, (Proc. Workshop, CMU, Pittsburg, June 30 – July 2, 1987), W. Sieg, (ed.), *Contemp. Math.*, **106** (1990) 181–196.
- [68] J. E. Hopcroft and J. D. Ullman, *Formal Languages and their Relations to Automata*, (Addison-Wesley, Reading, Mass, 1969).

- [69] C. G. Jockusch Jr., Ramsey's theorem and recursion theory, *J. Symbolic Logic*, **37** (1972) 268–280.
- [70] C. G. Jockusch Jr., A. A. Lewis and J. B. Remmel, Π_1^0 -classes and Rado's selection principle, *J. Symbolic Logic*, **56** (1991) 684–693.
- [71] C. G. Jockusch Jr. and T. G. McLaughlin, Countable retracing functions and Π_2^0 predicates, *Pacific J. Math.*, **30** (1969) 67–93.
- [72] C. G. Jockusch Jr. and R. A. Shore, Pseudo-jump operators, II. Transfinite iterations, hierarchies and minimal covers, *J. Symbolic Logic*, **49** (1984) 1205–1236.
- [73] C. G. Jockusch Jr. and R. I. Soare, A minimal pair of Π_1^0 classes, *J. Symbolic Logic*, **36** (1971) 66–78.
- [74] C. G. Jockusch Jr. and R. I. Soare, Degrees of members of Π_1^0 classes, *Pacific J. Math.*, **40** (1972) 605–616.
- [75] C. G. Jockusch Jr. and R. I. Soare, Π_1^0 classes and degrees of theories, *Trans. Amer. Math. Soc.*, **173** (1972) 33–56.
- [76] C. G. Jockusch Jr. and R. I. Soare, Boolean algebras, Stone spaces, and the iterated Turing jump, *J. Symbolic Logic*, **59** (1994) 1121–1138.
- [77] H. A. Kierstead, An effective version of Dilworth's theorem, *Trans. Amer. Math. Soc.*, **268** (1981) 63–77.
- [78] H. A. Kierstead, Recursive colorings of highly recursive graphs, *Canad. J. Math.*, **33** (1981) 1279–1290.
- [79] H. A. Kierstead, An effective version of Hall's theorem, *Proc. Amer. Math. Soc.*, **88** (1983) 124–128.
- [80] H. A. Kierstead, Recursive ordered sets, in: *Combinatorics and Ordered Sets*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf., Humboldt State Univ., Arcata, CA, Aug. 11–17, 1985), I. Rival, (ed.), *Contemp. Math.*, **57** (1986) 75–102.
- [81] H. A. Kierstead, Recursive and on-line graph coloring, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 1233–1269.

- [82] H. A. Kierstead, G. McNulty and W. T. Trotter Jr., A theory of recursive dimension for ordered sets, *Order*, **1** (1984) 67–82.
- [83] H. A. Kierstead and W. T. Trotter Jr., An extremal problem in recursive combinatorics, in: *Combinatorics, Graph Theory and Computing*, Vol. 2, (Proc. 12th. S. E. Conf., Baton Rouge, LA, 1981, Congr. Numer., **33** (1981) 143–153.
- [84] S. C. Kleene, Recursive predicates and quantifiers, *Trans. Amer. Math. Soc.*, **53** (1943) 41–73.
- [85] S. C. Kleene, A symmetric form of Gödel's theorem, *Nederl. Akad. Wetensch. Proc.*, **53** (1950) 800–802.
- [86] S. C. Kleene, Hierarchies of number-theoretic predicates, *Bull. Amer. Math. Soc.*, **61** (1955) 193–213.
- [87] J. F. Knight, A. H. Lachlan and R. I. Soare, Two theorems on degrees of models of true arithmetic, *J. Symbolic Logic*, **49** (1984) 425–436.
- [88] K. Ko, Complexity theory of real functions, *Progr. Theoret. Comput. Sci.*, (1991).
- [89] K. Ko, On the computability of fractal dimensions and julia sets, *Ann. Pure Appl. Logic*, (to appear).
- [90] D. König, Sur les correspondances multivoques des ensembles, *Fund. Math.*, **9** (1926) 114–134.
- [91] G. Kreisel, A variant to Hilbert's theory of the foundations of arithmetic, *British J. Philos. Sci.*, **4** (1953) 107–129.
- [92] G. Kreisel, Analysis of the Cantor-Bendixson theorem by means of the analytic hierarchy, *Bull. Acad. Polon. Sci. Sér. Sci. Math. Astronom. Phys.*, **7** (1959) 621–626.
- [93] A. Kučera, An alternative, priority-free, solution to Post's problem, in: *Math. Foundations of Comput. Sci.*, (Proc. Conf., Bratislava, Aug. 25–29, 1986), J. Grusha, B. Rován and J. Wiedermann, (eds.), *Lecture Notes in Comput. Sci.*, **233** (1986) 493–500.

- [94] D. Lacombe, Les ensembles récursivement ouverts ou fermés, et leurs applications à l'analyse récursive, *C. R. Acad. Sci. Paris*, **245** (1957) 1040–1043; **246** (1958) 28–31.
- [95] S. Lempp, Hyperarithmetical index sets in recursion theory, *Trans. Amer. Math. Soc.*, **303** (1987) 559–583.
- [96] F. W. Levi, Arithmetische Gesetze im gebiete diskreter gruppen, *Rend. Circ. Mat. Palermo*, **35** (1913), 225–236.
- [97] J. Lloyd, *Foundations of Logic Programming*, Symbol. Comput. Artificial Intelligence, (Springer-Verlag, Berlin and NY, 1st. edn. 1984, 2nd. edn. 1987).
- [98] D. McDermott, Nonmonotonic logic II: nonmonotonic modal theories, *J. Assoc. Comput. Mach.*, **29** (1982) 33–57.
- [99] D. McDermott and J. Doyle, Nonmonotonic logic I, *Artificial Intelligence*, **13** (1980) 41–72.
- [100] A. B. Manaster and J. G. Rosenstein, Effective matchmaking (recursion theoretical aspects of a theorem of Philip Hall, *Proc. London Math. Soc.*, Ser. 3, **25** (1972) 615–654.
- [101] A. B. Manaster and J. G. Rosenstein, Effective matchmaking and k -chromatic graphs, *Proc. Amer. Math. Soc.*, **39** (1973) 371–378.
- [102] V. W. Marek, A. Nerode and J. B. Remmel, A theory of nonmonotonic rule systems I, *Ann. Math. Artificial Intelligence*, **1** (1990) 241–273; [also in: *Logic in Computer Science*, (Proc. 5th. Annual IEEE Sympos. Logic in Comput. Sci., Philadelphia, PA, June 4–7, 1990), LICS–90, (IEEE Comput. Sci. Press, Los Alamitos, CA, 1990), 79–94].
- [103] V. W. Marek, A. Nerode and J. B. Remmel, A theory of nonmonotonic rule systems II, (Special issue: Artificial Intelligence and Mathematics, I, *Proc. 1st. Internatl. Sympos.*, Fort Lauderdale, FL, Jan. 3–5, 1990), M. C. Golumbic and F. Hoffman, (eds.), *Ann. Math. Artificial Intelligence*, **5** (1992) 229–263.
- [104] V. W. Marek, A. Nerode and J. B. Remmel, How complicated is the set of stable models of a recursive logic program?, *Ann. Pure Appl. Logic*, **56** (1992) 119–135.

- [105] V. W. Marek, A. Nerode and J. B. Remmel, A context for belief revision: forward chaining-normal nonmonotonic rule systems, *Ann. Pure Appl. Logic*, **67** (1994) 269–323.
- [106] V. W. Marek, A. Nerode and J. B. Remmel, The stable models of a predicate logic program, *J. Logic Programming*, **21** (1994) 129–154.
- [107] V. W. Marek, A. Nerode and J. B. Remmel, Nonmonotonic rule systems with recursive sets of restraints, (Special issue: Proc. Sacks Sympos., Cambridge, Mass., 1993), *Arch. Math. Logic*, **36** (1997) 339–384.
- [108] D. Marker, Degrees of models of true arithmetic, in: *Logic Colloq. '81*. (Proc. Herbrand Sympos., Marseilles, July, 16–24, 1981), J. Stern (ed.), *Stud. Logic Found. Math.*, **107** (1982) 233–242.
- [109] D. A. Martin and M. B. Pour-El, Axiomatizable theories with few axiomatizable extensions, *J. Symbolic Logic*, **35** (1970) 205–209.
- [110] G. Metakides and A. Nerode, Recursion theory and algebra, in: *Algebra and Logic*, (Proc. 14th. Summer Res. Inst. Austral. Math. Soc., Monash Univ., Clayton, Vic., Australia, Jan. 6 – Feb. 16, 1974), J. N. Crossley, (ed.), *Lecture Notes in Math.*, **450** (1975) 209–219.
- [111] G. Metakides and A. Nerode, Recursively enumerable vector spaces, *Ann. Math. Logic*, **11** (1977) 147–171.
- [112] G. Metakides and A. Nerode, Effective content of field theory, *Ann. Math. Logic*, **17** (1979) 289–320.
- [113] M. Minsky, A framework for representing knowledge, in: *The Psychology of Computer Vision*, P. H. Winston, (ed.), (McGraw Hill, NY, Auckland, Düsseldorf, 1975), 211–277.
- [114] Y. N. Moschovakis, Descriptive Set Theory, *Stud. Logic Found. Math.*, **100** (1980).
- [115] A. Nerode and W. Q. Huang, An application of pure recursion theory to recursive analysis (Chinese), *Acta Math. Sinica*, **28** (1985) 625–636.
- [116] A. Nerode and J. B. Remmel, A survey of lattices of r.e. substructures, in: *Recursion Theory*, (Proc. AMS-ASL Summer Inst., Ithaca, NY, June 28 – July 16, 1982), A. Nerode and R. A. Shore, (eds.), *Proc. Sympos. Pure Math.*, **42** (1985) 323–375.

- [117] A. Nerode and J. B. Remmel, Complexity theoretic algebra I: vector spaces over finite fields, in: *Structure in Complexity Theory*, (Proc. 2nd. Annual IEEE Conf. Struct. Complexity Theory, Cornell Univ., Ithaca NY, June 16–19, 1987), (IEEE Comput. Soc. Press, Washington, DC, 1987), 218–239.
- [118] A. Nerode and J. B. Remmel, Complexity-theoretic algebra II: Boolean algebras, (Special issue: 3rd. Asian Conf. Math. Logic, Beijing, Oct. 26–30, 1987), D. P. Yang, (ed.), *Ann. Pure Appl. Logic*, **44** (1989) 71–99.
- [119] A. Nerode and J. B. Remmel, Complexity-theoretic algebra: vector space bases, in: *Feasible Mathematics*, (Proc. Workshop, June 16–18, 1989, Cornell Univ. Ithaca, NY), S. Buss and P. J. Scott (eds.), *Progr. Comput. Sci. Appl. Logic*, **9** (1990) 293–319.
- [120] A. Nerode and J. B. Remmel, On the lattices of NP-subspaces of a polynomial time vector space over a finite field, *Ann. Pure Appl. Logic*, **81** (1996) 125–170.
- [121] P. Odifreddi, *Classical Recursion Theory. The Theory of Functions and Sets of Natural Numbers*, *Stud. Logic Found. Math.*, **125** (1989).
- [122] V. P. Orevkov, A constructive mapping of a square onto itself displacing every constructive point (Russian), *Dokl. Akad. Nauk*, **152** (1963) 55–58; [translated in: *Soviet Math. – Dokl.*, **4** (1963) 1253–1256].
- [123] M. B. Pour-El and J. I. Richards, *Computability in Analysis and Physics*, *Perspect. in Math. Logic*, (1989).
- [124] M. O. Rabin, Computable algebra, general theory and theory of computable fields, *Trans. Amer. Math. Soc.*, **95** (1960) 341–360.
- [125] R. Rado, Axiomatic treatment of rank in infinite sets, *Canad. J. Math.*, **1** (1949) 337–343.
- [126] R. Reiter, A logic for default reasoning, *Artificial Intelligence*, **13** (1980) 81–132.
- [127] J. B. Remmel, Recursively enumerable Boolean algebras, *Ann. Math. Logic*, **15** (1978) 75–107.
- [128] J. B. Remmel, On the effectiveness of the Schröder-Bernstein theorem, *Proc. Amer. Math. Soc.*, **83** (1981) 379–386.

- [129] J. B. Remmel, Graph colorings and recursively bounded Π_1^0 -classes, *Ann. Pure Appl. Logic*, **32** (1986) 185–194.
- [130] J. B. Remmel, Recursive Boolean algebras, in: *Handbook of Boolean Algebras*, Vol. 3, J. D. Monk, R. Bonnet and S. Koppelberg, (eds.), (North-Holland, Amsterdam, NY, 1989), Ch. 25, 1097–1165.
- [131] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, (1st. edn., McGraw-Hill, New York-Toronto, Ont.-London, 1967; 2nd. edn., MIT Press, Cambridge, Mass., London, 1987).
- [132] J. G. Rosenstein, Linear orderings, *Pure Appl. Math.*, **98** (1982).
- [133] J. B. Rosser, Extensions of some theorems of Gödel and Church, *J. Symbolic Logic*, **1** (1936) 87–91.
- [134] J. H. Schmerl, Recursive colorings of graphs, *Canad. J. Math.*, **32** (1980) 821–830.
- [135] J. H. Schmerl, The effective version of Brooks' theorem, *Canad. J. Math.*, **34** (1982) 1036–1046.
- [136] J. R. Shoenfield, Degrees of formal systems, *J. Symbolic Logic*, **23** (1958) 389–392.
- [137] J. R. Shoenfield, Degrees of models, *J. Symbolic Logic*, **25** (1960) 233–237.
- [138] J. R. Shoenfield, *Mathematical Logic*, (Addison-Wesley, 1967).
- [139] S. G. Simpson, Degrees of unsolvability: a survey of results, in: J. Barwise, (ed.), *Handbook of Mathematical Logic*, *Stud. Logic Found. Math.*, **90** (1977) 631–652.
- [140] R. L. Smith, Effective aspects of profinite groups, *J. Symbolic Logic*, **46** (1981) 851–863.
- [141] R. I. Soare, Cohesive sets and recursively enumerable Dedekind cuts, *Pacific J. Math.*, **31** (1969) 215–231.
- [142] R. I. Soare, Recursion theory and Dedekind cuts, *Trans. Amer. Math. Soc.*, **140** (1969) 271–294.

- [143] R. I. Soare, Recursively Enumerable Sets and Degrees. A Study of Computable Functions and Computably Generated Sets, *Perspect. in Math. Logic*, (1987).
- [144] D. R. Solomon, Reverse Mathematics and Ordered Groups, Ph.D. Thesis, Cornell Univ., Ithaca, NY, (1998).
- [145] E. Specker, Der Satz vom Maximum in der rekursiven Analysis, in: *Constructivity in Mathematics*, (Proc. Colloq., Amsterdam, Aug. 26–31, 1957), A. Heyting, (ed.), *Stud. Logic Found. Math.*, (1959) 254–265.
- [146] A. Tarski, *Logic, Semantics, Metamathematics, Papers from 1923 to 1938*, (Oxford Univ. Press, 1956).
- [147] J. J. Thurber, Recursive and r.e. quotient Boolean algebras, *Arch. Math. Logic*, **33** (1994) 121–129.

Douglas Cenzer
Department of Mathematics
University of Florida
Gainesville, FL 32611, USA
e-mail: cenzer@math.ufl.edu

Jeffrey B. Remmel
Department of Mathematics
University of California at San Diego
La Jolla, CA 92093, USA
e-mail: jremmel@ucsd.edu

This Page Intentionally Left Blank

Chapter 14

Computability Theory and Linear Orderings

R. G. Downey*

Contents

- Introduction
- 1. Preliminaries
- 2. Presentations
- 3. Computable isomorphisms
- 4. Theorems of Watnick and of Lerman
- 5. Effective content of some classical theorems:
subsequences, embeddings and automorphisms
- 6. Linear extensions of partial orderings, Dilworth's theorem,
and effective dimension
- 7. Degree-theoretical results, jump degrees
and worker type arguments
- 8. Other work and related results
- References

Introduction

The goal of this article is to give a survey of results and techniques in a fascinating area of algorithmic combinatorics. Computability theory, or recursion

*Author's research partially supported by grants from the IGC at Victoria University, by the MSRI at Berkeley and by the US/NZ cooperative science foundation. He wishes to thank Jeff Remmel and Bob Soare for helpful discussions. He wishes to thank Peter Donelan for much needed help with the overall presentation, and to thank Carl Jockusch and Michael Moses who supplied extensive corrections and revisions. Final preparation and rewriting was supported by the Marsden Fund for basic science in New Zealand.

theory¹, is concerned with our attempts to understand the algorithmic nature (or lack thereof) of the universe; combinatorics — particularly finite combinatorics — is concerned with counting, sorting and matching arguments. These areas combine particularly well, and the resulting theory is quite deep.

The main idea behind this area of computable combinatorics is the following. When we deal with ‘real’ structures in everyday mathematics, we would deal with objects such as the integers whose existence is somehow concrete. Similarly if we argue that a certain feature exists — say a solution to an equation — it is desirable to be able to give an algorithmic proof and hence *exhibit* a solution. In computable mathematics, this intuition can be formalised by asking that our structures be given in some algorithmic manner, and then examining the algorithmic content of such structures. In a loose sense one can view this as a theory of interactive programming where one feeds in an algorithm for the structure, and asks whether one can derive algorithms for various features of the structure, or else prove that this is impossible. Classical instances of such questions are the word problem of groups, Hilbert’s tenth problem, the decidability of the elementary theory of abelian groups, and the four-colour problem. In the word problem we are given a group by a ‘finite presentation’ i.e.,

$$G = \langle x_1, \dots, x_n : y_1, \dots, y_m \rangle$$

where $\langle x_1, \dots, x_n \rangle$ is a free group and $y_1, \dots, y_m$ generate a free normal subgroup of $\langle x_1, \dots, x_n \rangle$. One then asks if there is an algorithm to decide if a word in $x_1, \dots, x_n$ is the identity or not in G . A famous result of Novikov [178] and Boone [26] shows that it is possible to select G so that the answer is no. Hilbert’s tenth problem asks whether we can decide if a given polynomial (in many variables) over $\mathbb{N}$ has positive integer solutions. Again the answer is no (Matijasevic [152]). The question on abelian groups is whether given a simple formula $\varphi(x_1, \dots, x_n)$ in the language of groups, we can decide if $\varphi(x_1, \dots, x_n)$ is true of all abelian groups. Here the answer is that an algorithm does indeed exist (W. Szmielew [235]). Finally we know by Appel and Haken that any finite planar graph is 4-colourable. The question

¹The study of the effective content of mathematics has been referred to in many ways. Following early work of Kleene it has been known as “recursion theory.” However, we prefer to use the name “computability theory” which captures the original spirit and intentional meaning of the area. We refer the reader to Soare [233] for a discussion of these issues.

is how fast can this be done? Recently Appel and Haken have found a fast (polynomial time) algorithm to do this.

In our studies we seek to understand the relative difficulty of such problems.

At this point we should stress that we are concerned with studying these problems in the area of classical mathematics and not as ‘constructivists’ whose philosophical outlook is to regard as not existing anything which cannot be ‘theoretically’ actually constructed. As we shall see, there are also some related questions in so-called ‘reverse mathematics’. This area seeks to classify problems proof-theoretically (according to how much ‘comprehension’ is needed) rather than algorithmically. (see e.g., Friedman-Simpson-Smith [76]).

In the present paper, we shall study the computability theory of linear orderings. There is, of course, a lot of other work on computable combinatorics which we shall not examine (although we do give some brief comments in Section 8). Bill Gasarch has a survey article devoted to this area in the present volume.

We hoped to make the paper accessible to a (reasonably determined) combinatorialist who was only noddingly acquainted with computability theory. Thus in Section 1 we give a brief account of the basics of computability theory needed for this article. Some proofs are sketched to give the reader a flavour of the types of arguments we need. This section might also serve as a reminder to those logicians who work in areas other than computability theory. Experts should jump immediately to Section 2.

In Section 2 we analyse the notion of presentation of a structure (ordering). This generalises the notion of presentation of a group (alluded to above) and is crucial to the remainder of the paper. The key tension is between the notions of *effective* invariant and *classical* invariant. Here we investigate results such as those of Chisholm and Moses [32], Feiner [66, 67], Rosenstein [208], Richter [201]. Understanding the degrees of presentability has been an area of intensive research and we refer the reader to Downey [51] for further details here.

In Section 3 we consider how computable properties may partition classical isomorphism types of computable orderings. We begin by leading up to Remmel’s [193] characterisation of computably categorical linear orderings as those with finitely many successivities. We then turn to the results of Moses [167] and Schwartz [219] on more restrictive classes such as 1-computable and decidable orderings.

In Section 4 we introduce the infinite injury priority method and the fundamental results of Watnick [243] and Lerman [138]. We give a previously unpublished proof (due to the author) of Watnick's [243] result on finite condensations of order types. For those familiar with the standard 'tree-of-strategy' Π_2 arguments, it is hoped that this will be much more accessible than either Watnick's [243] or Ash-Jockusch-Knight's [9] version. (No knowledge of Π_2 arguments is assumed here. The main thrust of this chapter is trying to understand the complexity of distinguished suborderings of an ordering. For instance, Watnick's theorem asserts that τ is an order type with a Π_2 copy iff $\zeta\tau$ has a computable copy. Here ζ denotes the order type of the integers). We also present in this section some related work of Fellner [71], Rosenstein, Moses and the author.

In Section 5 we look at the effective content of various classical theorems, concentrating on embeddings and automorphisms. Here, by effective content we mean the following. Consider the result that every infinite linear ordering has an infinite ω or ω^* sequence (here ω denotes the order type of $\mathbb{N}$ and ω^* its reversal). We ask is this true effectively? That is, does each *effective* infinite linear ordering have an *effective* ω or ω^* sequence? An old theorem of Tennenbaum [unpubl.] and independently Denisov [unpubl.] says the answer is no in a very strong way. So in this section we concentrate on extensions of the Tennenbaum-Denisov result by Watnick, Hird, Lerman and others, and other related questions on automorphisms and distinguished subsequences. Of course, if a theorem fails to hold effectively, we seek effective versions. For instance, as we see in this section, Rosenstein showed that if R is a computable ordering then R has a computable subsequence of order type ω , ω^* , $\omega + \omega^*$ or $\omega + \zeta\eta + \omega^*$ (here η denotes the order type of the rationals) and Lerman proved that all of these types are necessary. Another effective version of the same theorem is Manaster's result that R must have a Π_1 ω or ω^* sequence. Thus there can be many effective versions of the same theorem. We also look at the effective content of the Dushnik-Miller theorem on self embeddings, and finally look at the Downey - Moses and Schwartz results on automorphisms and self embeddings.

In Section 6 we look at linear extensions of computable partial orderings and so the effective content of Szpilrajn's [236] result that any *poset has a linear extension*; which leads to the well-known theory of dimension for posets. Here we also analyse the Slaman-Woodin result of dense linear extensions. We look at Kierstead's beautiful results on computable covering numbers for Dilworth's theorem and the theory of computable dimension developed

by Kierstead, Rosenstein, McNulty, Schmerl, Trotter and others. Finally we also look at the Manaster-Rosenstein-Remmel theory of computable 2-dimensional partial orderings. The results of this section are certainly not well known by computability theorists and really deserve to be, as they form a very attractive theory with many difficult open questions.

In Section 7 we study degree theoretical aspects of presentations, and arguments of the ‘worker types’ whose roots go back to work of Harrington on Peano arithmetic. For instance we look at the Jockusch-Soare [107] result that there are low linear orderings not isomorphic to computable ones and give a new proof of the Downey-Moses [59] result on degrees of successivities in linear orderings. We finish this section with the Ash-Jockusch-Knight [9] and Downey-Knight [55] results on jump degrees of orderings. (The arguments here are computability-theoretically very sophisticated and are mainly only very briefly sketched.)

In the last section (Section 8) we briefly discuss other areas of computable combinatorics, and other areas of computable aspects of linear orderings.

The philosophy of the paper is to give at least one example of all of the techniques of proof. There are a couple of exceptions where, for brevity, we have omitted a result that would take us too far afield (e.g., Knight’s proof that if A has proper 1-degree then that degree is $\mathbf{0}'$, which uses forcing) and in other cases we have been quite sketchy in the interests of keeping this paper of finite length and concentrating on the *ideas*. In the case of the Kierstead-McNulty-Trotter-Rosenstein results in Section 6, there is also a very nice survey by Kierstead [120] and McNulty [155] that the reader is urged to read. (We have tried to avoid the game-theoretical language used by these authors and this makes our treatment rather different from either of these surveys.)

Finally we remark that there are quite a few new results, unpublished results and new proofs of old results included here, and numerous open questions scattered throughout. In the case of open questions, we have tried to find out exactly to whom they were due but in many cases they seem to belong to folklore.

Notation is standard. We let ω , ω^* , ζ , η denote the order types of, respectively, $\mathbb{N}$, $-\mathbb{N}$, $\mathbb{Z}$, and $\mathbb{Q}$. An order type is called *scattered* if it does not embed η as a subordering. We define the standard sum $\alpha + \beta$ and product $\alpha \cdot \beta$ of orderings via: $\alpha + \beta$ results from putting a copy of α followed by one of β , and $\alpha \cdot \beta$ results from replacing each point in an ordering of type β by one of type α . We say that (x, y) is called a *successivity* or *adjacency* in an

ordering A if $(\forall z \in A)(z \notin (x, y))$. The *width* of a poset is the size of the largest antichain. We let $\langle \cdot, \cdot \rangle$ denote a standard bijection from $\mathbb{N} \times \mathbb{N}$ to $\mathbb{N}$. We let $2^{<\omega}$ denote the tree of all binary sequences and 2^ω the collection of all paths through $2^{<\omega}$. Any other needed definitions or terminology will be introduced in the paper.

1 Preliminaries

Many mathematicians unfortunately do not have a training in recursion theory. Hopefully this section will give a reasonably self-contained account of some of the basics in this area. For more details the reader is referred to, for example, Salomaa [215], Rogers [207], Soare [232], or Davis and Weyuker [43].

Our concern is with functions from $A \rightarrow B$ where $A, B \subseteq \mathbb{N}$; i.e., *partial* functions on $\mathbb{N}$. If $A = \mathbb{N}$ with the function is called *total*. Looking only at $\mathbb{N}$ may seem rather restrictive, but remember we are dealing with algorithmic structures, and all such structures (from our point of view) can be coded as subsets of $\mathbb{N}$. For example, if we consider the rationals $\mathbb{Q}$, these can be considered as coded in $\mathbb{N}$ as follows:

Let $r \in \mathbb{Q} - \{0\}$; write $r = (-1)^\delta \left(\frac{p}{q}\right)$ with $p, q \in \mathbb{N}$ in lowest terms and $\delta = 0$ or 1 . Then define the Gödel number of r , $\#(r)$, as $2^\delta 3^p 5^q$, with the Gödel number of 0 to be 0 .

Then by the fundamental theorem of arithmetic, $\#$ describes an injection from $\mathbb{Q}$ into $\mathbb{N}$ and furthermore given $n \in \mathbb{N}$ we can decide exactly which $r \in \mathbb{Q}$, if any, has $\#(r) = n$. Similarly if σ is a finite binary string, say $\sigma = a_1, a_2, \dots, a_n$ then we can define $\#(\sigma) = 2^{a_1+1} 3^{a_2+1} \dots (P_n)^{a_n+1}$ where P_n denotes the n -th prime.

The above procedures are called ‘effective coding’ since they give an algorithm for the relevant injection.

Speaking of algorithms, what exactly do we mean? Here we return to the celebrated Church-Turing Thesis. Paraphrased in modern terminology, the Church-Turing Thesis states that *the collection of algorithmic partial functions are exactly those that can be computed by Pascal programmes*. Of course, neither Turing nor Church said it like this. Rather, there are many formulations of the class of computable functions — via Turing machines, λ -calculus, Kleene schemes, etc., but *they all turn out to be equivalent*. We will call them

the *partial recursive functions* or the *computable partial functions*. If a partial recursive function is total we shall simply call it *recursive* or *computable*. For our purposes, it is really not necessary to describe them in detail, but we will need to know two very important properties of this class of functions:

Property 1.1 (Enumeration Theorem – Universal Turing Machine) *There is an algorithmic or computable way of enumerating all the computable partial functions. That is, there is a computable partial function $f(x, y)$ of two variables such that*

$$f(x, y) = \varphi_x(y)$$

where $\varphi_x(y)$ denotes the x -th computable partial function on input y ; and this makes sense.

Sketch Proof. To see that property 1.1 holds, again we can return to Gödel numbering. A programme is just a finite set of lines with a finite set of possible statements on each line (more or less). We can assign to statements such as ‘else’, ‘go to’ etc. Gödel numbers, so that each line could be assigned a number. For example, if GO TO were assigned to 50,

15 GO TO 25

could be numbered as

$$3^{\#(15)} 5^{\#(50)} 7^{\#(25)}.$$

We do this so that each line has a unique number and then code up the programme as

$$2^{\#(l_1)} 3^{\#(l_2)} \dots (P_n)^{\#(l_n)}$$

where $\#(l_i)$ denotes the Gödel number of l_i .

In this way we can clearly establish property 1.1. □

The *point* of property 1.1 is that we can *pretend* that we have all the machines $\varphi_1, \varphi_2, \dots$ in front of us; to compute 10 steps in the computation of the 3rd machine on input 20, we can pretend to walk to the 3rd machine, put 20 on the tape and run it for 10 steps (we write this as $\varphi_3^{10}(20)$). In many ways, property 1.1 is the platform that makes undecidability proofs work since it allows us to diagonalise *over* the class of computable partial functions *without leaving the class*. For instance

Property 1.2 (Unsolvability of the Halting problem) *There is no algorithm which given x, y decides if $\varphi_x(y) \downarrow$, that is, if the x -th programme on input y halts. Indeed there is no algorithm to decide if $\varphi_x(x) \downarrow$.*

Proof. Suppose such an algorithm exists. Then by property 1.1, it follows that the following function g is (total) computable:

$$g(x) = \begin{cases} 1 & \text{if } \varphi_x(x) \uparrow \quad (\text{i.e., } \varphi_x(x) \text{ does not halt}) \\ \varphi_x(x) + 1 & \text{if } \varphi_x(x) \downarrow \end{cases}$$

Again using property 1.1, there is a y with $g = \varphi_y$. As $g(y) \downarrow$, $\varphi_y(y) \downarrow$ and hence $g(y) = \varphi_y(y) + 1 = g(y) + 1$, a contradiction. $\square$

Note that we *can* define a computable *partial* g via $g(x) = \varphi_x(x) + 1$ and avoid contradiction as it will follow that, for any *index* y of g , $\varphi_y(y) = g(y) \uparrow$. (In fact such a g has no computable (total) extension (exercise).) Also the reason for the use of *partial* computable functions in property 1.1 is clear. The argument above will show that there is no computable procedure to generate all (and only) the total computable functions. Actually the result above can be used to show many problems are algorithmically unsolvable by ‘coding’ the halting problem. For example

Proposition 1.1 *There is no algorithm to decide whether $\text{dom } \varphi_x$ is empty.*

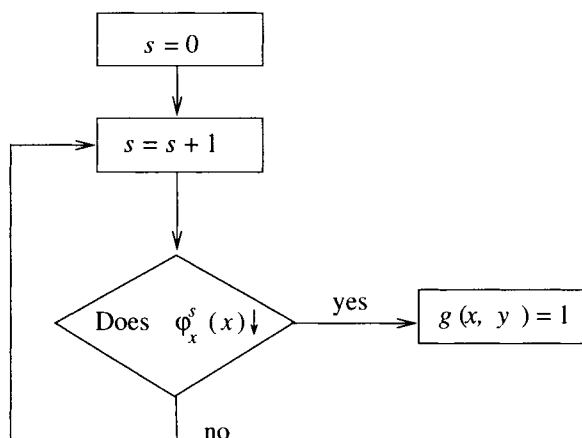
Proof. We code the halting problem into the problem of deciding whether $\text{dom } \varphi_x = \emptyset$. That is, we show that *if* we could decide whether $\text{dom } \varphi_x = \emptyset$ *then* we could solve the halting problem. Define a computable partial function via, for all $y \in \mathbb{N}_0$,

$$g(x, y) = \begin{cases} 1 & \text{if } \varphi_x(x) \downarrow \\ \uparrow & \text{if } \varphi_x(x) \uparrow. \end{cases}$$

To see that g is computable partial, compute $g(x, y)$ using the flowchart below.

Now we can consider $g(x, y)$ as a computable collection of computable partial functions (this is called the *s-m-n theorem*). The *s-m-n* theorem asserts there is a computable $s(x)$ such that, for all x, y

$$\varphi_{s(x)}(y) = g(x, y).$$



Then

$$\text{dom}(\varphi_{s(x)}) = \begin{cases} \mathbb{N} & \text{if } \varphi_x(x) \downarrow \\ \emptyset & \text{if } \varphi_x(x) \uparrow. \end{cases}$$

So if we could decide for a given x if $\varphi_{s(x)}$ has empty domain, we could solve the halting problem. $\square$

The reasoning used in proposition 1.1 can be pushed a lot further. First we can regard problems as coded by subsets of $\mathbb{N}$. For example, the halting problem is coded by $K = \{x : \varphi_x(x) \downarrow\}$ (or, for example, $K^* = \{2^x 3^y : \varphi_x(y) \downarrow\}$). We need some terminology.

Definition 1.1 A set $A \subseteq \mathbb{N}$ is called

- (i) *computably enumerable* (r.e) if $A = \text{dom } \varphi_x$ for some x .
- (ii) *computable* if A and $\bar{A} (= \mathbb{N} - A)$ are both computably enumerable.

We will let W_e denote the e -th computably enumerable set. That is, we let $W_e = \text{dom } \varphi_e$ and let $W_{e,s} = \{x \leq s : \varphi_e^s(x) \downarrow\}$ constitute s steps in the enumeration of W_e .

The name *computably enumerable* comes from a notion of ‘effectively countable’ via the following characterisation.

Proposition 1.2 *An infinite set A is computably enumerable iff there is a (total) computable injective function f with $f(\mathbb{N}) = A$.*

Proof. ($\Rightarrow$) Suppose $A = \text{dom } \varphi_x$. Compute f in stages. To compute $f(0)$, find the least s and then the least z for s such that $z \leq s$ and $\varphi_x^s(z) \downarrow$. Define $f(0) = z$. To compute $f(1)$ find the least $s_1 \geq s$ and the least $z \neq z_1 \leq s_1$ such that $\varphi_x^{s_1}(z_1) \downarrow$. $f(s_1) = z_1$, and similarly we define $f(j)$ at step j to be the least element of $\text{dom } \varphi_x^{s_j}$ not yet in the range of f . Clearly f is computable and injective.

($\Leftarrow$) Let $f(\mathbb{N}) = A$ as in the hypothesis of the theorem. Define $g^s(x) = 1$ iff $f(s) = x$. Then g is clearly computable partial and $\text{dom } g = \text{ran } f = A$. $\square$

Thus we can think of an computably enumerable set as an effective infinite list. (But **not** *computably in order*.) Note that computable sets correspond to decidable questions; for if A is computable, then to decide if $x \in A$, let $f(\mathbb{N}) = A$ and $g(\mathbb{N}) = \bar{A}$. Now enumerate $f(0), g(0), f(1), g(1), \dots$ until x occurs (as it must) in the A or $\bar{A}$ list. Note also that property 2 states that $K = \{x : \varphi_x(x) \downarrow\}$ is a computably enumerable but not computable set.

An *index set* is a set A such that if $x \in A$ and $\varphi_x = \varphi_y$ then $y \in A$. Most of the sets we have looked at are index sets (e.g., $\{x : \text{dom } \varphi_x = \emptyset\}$ is an index set). Generalising proposition 1.1 we have

Theorem 1.3 (Rice) *An index set A is computable (and so the problem it codes is decidable) iff $A = \mathbb{N}$ or $A = \emptyset$.*

Proof. The proof of Rice's Theorem is very similar to the proof of 1.1.

Let $A \neq \mathbb{N}, \emptyset$ be an index set and without loss of generality, $e \in A$ where $\text{dom } \varphi_e = \emptyset$. Fix $z \in A$. Then for some q , $\varphi_z(q) \downarrow$. By the technique of proposition 1 (*s-m-n theorem*), there is a computable $s(x)$ such that, for all $y \in \mathbb{N}$,

$$\varphi_{s(x)}(y) = \begin{cases} \varphi_z(y) & \text{if } \varphi_x(x) \downarrow \\ \uparrow & \text{if } \varphi_x(x) \uparrow. \end{cases}$$

Then $\varphi_x(x) \downarrow$ implies $\varphi_{s(x)} = \varphi_z$ and so $s(x) \in A$, and $\varphi_x(x) \uparrow$ implies $\varphi_{s(x)} = \varphi_e$ and so $s(x) \notin A$. So if A were computable, K would also be computable. $\square$

Of course many decision problems are not coded by index sets and so can have decidable solutions.

The idea used in these proofs is that of *reducibility*. Namely, we use the strategy of “if we can do A then it allows us to do B ”. We can formalise this as follows.

We can regard one problem as being at least as hard as another by attaching to our computer an infinite read-only memory. We then say $A \leq_T B$ (A is Turing reducible to B) if we can compute A given that B is put in order into the memory, and we are allowed to ask for each x , a finite number of questions of the memory during the computation that decides whether a particular x is in A . This is a partial ordering. In the example above (proposition 1.1) let $E = \{x : \text{dom } \varphi_x = \emptyset\}$, we showed that $K \leq_T E$ by showing that $x \in K$ iff $s(x) \in E$. Indeed Rice’s theorem demonstrates that a nontrivial index set I will always have $K \leq_T I$. We will write $A \equiv_T B$ if $A \leq_T B$ and $B \leq_T A$. The equivalence classes give a notion of equicomputability and are called *degrees (of unsolvability)*. If $A \leq_T B$ we say A is B -computable. We let $\mathbf{0}$ denote the degree of the computable sets. We call machines with the infinite memory attached *oracle machines*. We can regard normal machines as oracle machines with empty memory.

We get the analogue of propositions 1.1 and 1.2 for oracle machines. That is

Property 1.3 *There is an enumeration of all oracle machines $\{\Phi_x : x \in \mathbb{N}\}$.*

Notation 1.1 We will let $\Phi_{e,s}(C;x)$ denote s stages in the computation of Φ_e with oracle C and input x .

Property 1.4 *If A is any set and $\Phi_x(A;y)$ denotes the result, if any, of computing the result of input y on the x -th oracle machine with oracle A (i.e., A in the memory), then $K^A = \{x : \Phi_x(A;x) \downarrow\}$ is not A -computable.*

We refer to K^A as ‘ K relativised to A ’, and the process of generalising a fact about computably enumerable sets to all oracles as *relativisation*. Note that $A <_T K^A$ (see below).

Remark 1.1 As we will see, we can also put resource bounds on our procedures. For example if we count steps and ask that computations halt in a polynomial (in the length of the input) number of steps we arrive at the polynomial computable functions, and the notion of polynomial time reducibility. We will discuss this more fully in Section 8.

If A is a set, K^A is a standard operator on A called the jump operator. We write A' for K^A , if the degree of A is $\mathbf{a}$ then we write $\mathbf{a}'$ for $\deg(A')$ (we use boldface letters for degrees). Since $B'T > B$ for all B , consequently we have defined a hierarchy of degrees

$$\mathbf{0}, \mathbf{0}', \mathbf{0}'', \dots$$

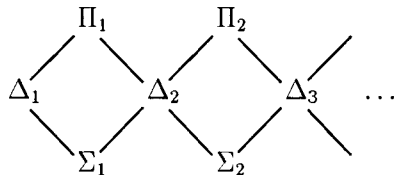
If $\mathbf{a} < \mathbf{0}^{(n)}$ for some n , we call $\mathbf{a}$ *arithmetical*. Note that since each degree is countable and has only countably many predecessors (only countably many machines), there are uncountably many degrees but only countably many arithmetical ones.

The name *arithmetical* comes from the following alternative methods of defining (e.g.) $\mathbf{0}'$. We define the notions of Σ_n , Π_n and Δ_n as follows:

We say a computable set is Δ_1 . A set B is called Σ_n if $x \in B$ holds iff there is a computable relation $R(x_1, \dots, x_n, x)$ with

$$\underbrace{\exists x_1 \forall x_2 \exists \dots}_{n \text{ alternations of quantifiers}} R(x_1, \dots, x_n, x)$$

We similarly say that a set Q is Π_n as we did for B except we now have the leading quantifier $\forall$ followed by n alternations of quantifiers. Note that we can always collapse two quantities of the form $\exists x_1 \exists x_2$ into a single one of the form $\exists x_3$ using Gödel numbers, which is why it makes sense to only look at *alternations* of quantifiers. Finally, we say a set R is Δ_n if it is both Σ_n and Π_n . With these notions we get the arithmetical hierarchy of Kleene



Here lines mean inclusion (rightward along the page), and all inclusions are proper. There is a strong relationship between the above and degrees. For instance

Theorem 1.4 (Kleene) *A set A is computably enumerable iff A is Σ_1 .*

Proof. Suppose A is computably enumerable. Then $A = \text{dom } \varphi_x$ for some x . Then $y \in A$ iff $(\exists s)(\varphi_x^s(y) \downarrow)$. Conversely, if A is Σ_1 , then for some computable R we have $y \in A$ iff $(\exists z)(R(z, y))$. Define a computable partial g by setting $g(y) = 1$ at stage s iff $s \geq y$ and $\exists z < s$ with $R(z, y)$ holding. $\square$

In terms of degree, $K = \emptyset'$ is the most complicated computably enumerable set. To see this, let $K_0 = \{\langle x, y \rangle : x \in W_y\}$ where $\langle \ , \ \rangle$ is a computable bijection from $\mathbb{N} \times \mathbb{N}$ to $\mathbb{N}$. Note that if W is computably enumerable then $W \leq_T K_0$ since $x \in W$ iff $\langle x, y \rangle \in K_0$ where $W = W_y$. As an example of a typical reduction, we show that $K_0 \equiv_T K$. First $K \leq_T K_0$ since K is computably enumerable. Now to see that $K_0 \leq_T K$, use the s - m - n theorem to define a computable class of functions:

$$\varphi_{s(\langle x, y \rangle)}^{(z)} = \begin{cases} 1 & \text{if } \varphi_x(y) \downarrow \quad (\text{i.e. } x \in W_y) \\ \uparrow & \text{otherwise} \end{cases} \quad (1.1)$$

Then $\langle x, y \rangle \in K_0$ iff $\varphi_x(y) \downarrow$ iff $\varphi_{s(\langle x, y \rangle)}(s(\langle x, y \rangle)) \downarrow$ iff $s(\langle x, y \rangle) \in K$. Hence $K_0 \leq_T K$.

It can be shown that a set $A \leq_T K$ iff A is Δ_2 and, more generally $A \leq_T \emptyset^{(n)}$ iff A is Δ_{n+1} . The arithmetical sets are those you ‘get from Peano arithmetic’ (roughly speaking).

To finish this section we will discuss a very important problem in the degrees: Post’s problem. By the above we see that all index sets are of degree $\geq \mathbf{0}'$. Post [185] observed that in 1944 all known computably enumerable problems had the property that they were either of degree $\mathbf{0}$ or $\mathbf{0}'$. He asked

Question 1.1 (Post’s Problem) *Does there exist a computably enumerable degree $\mathbf{a}$ with $\mathbf{0} < \mathbf{a} < \mathbf{0}'$?*

Post’s problem was finally solved by Friedberg [72] and Muchnik [172] using a new and ingenious method called the priority method that we shall meet in Section 2. They showed that

Theorem 1.5 (Friedberg, Muchnik) *There exist computably enumerable degrees $\mathbf{a} \mid \mathbf{b}$. That is, $\mathbf{a} \not\leq \mathbf{b}$ and $\mathbf{b} \not\leq \mathbf{a}$.*

There have been many extensions of the Friedberg-Muchnik theorem. Some notable extensions due to Sacks [213]: the computably enumerable degrees

are dense, and Lachlan [133] and Yates [244]: there exist computably enumerable degrees $\mathbf{a}$, $\mathbf{b} \neq \mathbf{0}$ with $\mathbf{a} \cap \mathbf{b} = \mathbf{0}$ and $\mathbf{c}$, $\mathbf{d}$ with $\mathbf{c} \cap \mathbf{d}$ not existing. Thus the computably enumerable degrees are a dense upper semilattice that is not a lattice and is not homogeneous. Soare [232] contains a lot of information on the computably enumerable degrees, computably enumerable sets and related structures.

We will not prove any of these results, and will look at the priority method later. There is however one crucial ingredient of the proof of the Friedberg-Muchnik theorem that we will use in some of the work to follow and which we will look at in this section.

Lemma 1.6 (Use principle) *Suppose $\Phi(A; x) \downarrow$. Let $u(\Phi(A; x))$ denote the maximum element x such that the question “is x in the memory?” is called during the computation of $\Phi(A; x)$. Let B be any set such that $B[u] = A[u]$, where for any set C , $C[z] = \{p : p \leq z \text{ and } p \in C\}$ and $u = u(\Phi(A; x))$. Then $\Phi(A; x) = \Phi(B; x)$.*

Proof. Both A and B give the same answers to the question in the computations, hence the result must be the same. $\square$

One use of Lemma 1.6 is an important characterisation of Δ_2 sets (i.e., sets $A \leq_T K$).

Lemma 1.7 (Shoenfield Limit Lemma) *A is Δ_2 iff there is a computable function $g(,)$ such that*

- (i) $\lim_s g(x, s)$ exists, i.e., $|\{s : g(x, s) \neq g(x, s+1)\}| < \infty$.
- (ii) $\lim_s g(x, s) = A(x)$ where we identify sets with their characteristic functions, i.e., $A(x) = 0$ if $x \notin A$ and $A(x) = 1$ if $x \in A$.

Proof. ($\Rightarrow$) Suppose A is Δ_2 . That is $A \leq_T \emptyset'$ so that for some procedure Φ_e we have $\Phi_e(K) = A$. Define $g(x, s) = 0$ if $\Phi_{e,s}(K_s; x) \uparrow$ or $\Phi_{e,s}(K_s; x) \downarrow \neq 1$ and $g(x, s) = 1$ otherwise.

(Here $K = \cup_s K_s$ is some computable enumeration of K .) Given x , let $u = u(\Phi_e(K; x))$. Let $s = s(x)$ be the stage where $K_s[u] = K[u]$. Let $t \geq s$ be the least stage such that $\Phi_{e,t}(K; x) \downarrow$. As $K_t[u] = K_s[u] = K[u]$ we have, by the use principle, $\Phi_{e,t}(K; x) = \Phi_{e,t}(K_t; x) = \Phi_{e,q}(K_q; x) = \Phi_e(K; x)$ for all $q \geq t$. Then for all $q \geq t$, $g(x, t) = g(x, q) = A(x)$ by definition.

($\Leftarrow$) Suppose such a function g exists. We construct an computably enumerable set B and a reduction Γ so that $\Gamma(B) = A$. Then $A \leq_T \emptyset'$ since A is computably enumerable (and so $A \leq_T \emptyset'$ by the fact that $K_0 \leq_T K$ in equation (1.1)).

We define $\Gamma(B; x)$ and B in stages, for all x .

STAGE 0:

Let $\gamma(x, 0) = \langle x, 0 \rangle$ and define $\Gamma_0(B_0; x) = g(x, 0)$ and $B_0 = \emptyset$.

STAGE $s + 1$:

We have defined $\gamma(x, s)$. If $g(x, s + 1) = g(x, s)$, then keep $\gamma(x, s) = \gamma(x, s + 1)$, $B_{s+1} = B_s$ and $\Gamma_{s+1}(B_{s+1}; x) = \Gamma_s(B_s; x)$. If $g(x, s + 1) \neq g(x, s)$, enumerate $\gamma(x, s)$ into B_{s+1} . Set $\Gamma_{s+1}(B_{s+1}; x) = g(x, s + 1)$. It will be the case that $\gamma(x, s) = \langle x, j \rangle$ for some $j \in \mathbb{N}$. Set $\gamma(x, s + 1) = \langle x, j + 1 \rangle$.

The construction succeeds as we now see. Note that $\lim_s \gamma(x, s) = \gamma(x)$ exists since $\gamma(x, s + 1) \neq \gamma(x, s)$ iff $\gamma(x, s + 1) \neq \gamma(x, s)$. Furthermore $\gamma(x, s + 1) \neq \gamma(x, s)$ iff $\gamma(x, s) \in B_{s+1} - B_s$.

Also, $\lim_s \Gamma_s(B_s; x) = \Gamma(B; x)$ since $\Gamma_s(B_s; x) \neq \Gamma_{s+1}(B_{s+1}; x)$ iff $\gamma(x, s) \in B_{s+1} - B_s$. It follows that Γ is B -computable. B can decide $\Gamma(B; x)$ ($= A(x)$) as follows. Given x and B , compute the least s such that $\gamma(x, s) \notin B$. Then $\Gamma_s(B_s; x) = \Gamma(B; x) = A(x) = g(x, s) = g(x)$ by construction. $\square$

Remark 1.2 Intuitively in the ($\Leftarrow$) part of the above proof K can answer $(\exists s)(g(x, s) \neq g(x, s + 1))$ and hence computably in K we can compute the limit of $g(x, s)$ and hence compute $A(x)$.

2 Presentations

Before we go any further we need to clarify what we mean by presenting a structure. First, all universes are identified with $\mathbb{N}$ under some coding, and hence we need only worry about the complexity of the ordering relation. We will write $\mathbf{A} = (\mathbb{N}, \leq_A)$ for an ordering and will write $\leq$ for $\leq_A$ when the context is clear.

Definition 2.1 A linear ordering $\mathbf{A} = (\mathbb{N}, \leq_A)$ is called *computably presented* if $\leq_A$ is a computable relation (on $\mathbb{N}$). [Equivalently, the open diagram (or a *computable ordering*) of $\mathbf{A}$ is computable.]

‘Effectivising’ the proof often attributed to Cantor, we see:

Theorem 2.1 *$\mathbf{A}$ is computably presented iff $\mathbf{A}$ is computably isomorphic to a computably subset of $\mathbb{Q}$ (under the usual ordering). We call this a computably subordering of $\mathbb{Q}$.*

Proof. Fix some Gödel numbering of $\mathbb{Q}$ which we denote by $\#$. Label $\mathbf{A}$ as $\{a_0, a_1, \dots\}$. At stage $s + 1$, we will have already defined $f(a_0), \dots, f(a_s)$. Now find the appropriate i and/or j such that $a_i < a_{s+1} < a_j$ and for all $k \leq s$, if $k \neq i, j$ then $a_k \notin (a_i, a_j)$. We can compute a rational r_{s+1} with $\#(r_{s+1}) > s + 1$, and $f(a_i) < r_{s+1} < f(a_j)$. Define $f(a_{s+1}) := r_{s+1}$.

It is easy to see that f is an embedding, and furthermore raf is a computably set (as $\text{raf} = B$ is increasing so $x \in B$ iff x is a member of B by stage (x)). $\square$

For the purposes of our investigations we’d like to extend Definition 2.1 and Theorem 2.1. We say A is a Σ_n^- (resp. Δ_n^- , Π_n^- , $\mathbf{a}^-$) *presented* linear ordering if $\leq$ is a Σ_n^- (resp. Δ_n^- , Π_n^- , degree $\mathbf{a}^-$) set.

If we relativise Theorem 2.1 we see that if $\mathbf{A}$ is $\mathbf{a}^-$ -presented, then $\mathbf{A}$ is isomorphic to a subordering of $\mathbb{Q}$ of degree $\mathbf{a}$. Suppose that $\mathbf{A}$ is Σ_1^- -presented. We claim that $\mathbf{A}$ is isomorphic to a Π_1 subordering of $\mathbb{Q}$. If $\mathbf{A}$ is Σ_1 then $x \leq y$ iff $(\exists z)R(x, y, z)$ holds. Since $\leq$ is linear we know at least one of $x \leq y$ or $y \leq x$ hold. By searching, given x and y we can decide one of $x \leq y$ or $y \leq x$ holding. The problem is that we can’t decide if $x = y$ holds (i.e., if **both** hold). Note that $x = y$ is Σ_1 , so there is a computably relation $\hat{R}$ such that $x = y$ iff $\exists s \hat{R}(x, t, s)$. Write $x \neq_s y$ if $\neg \hat{R}(x, y, s)$.

Now follow the proof of Theorem 2.1. In 2.1, at stage s , we need to work out all the $a_i \leq a_j$ relationships for $i, j < s$. Now, if we follow 2.1, the problem we need to resolve is that we see $a_i \leq a_j$ and $a_i \neq_s a_j$ so we define $f_s(a_i) = r_1$ and $f_s(a_j) = r_2$, say, with $r_1 \neq r_2$. Now at some stage $t > s$ it may be that $a_i =_t a_j$ (and hence for all $q \geq t$, $a_i =_q a_j$). Let $i < j$, then our action is to delete r_2 from raf by defining $f_t(a_i) = f_t(a_j) = r_1$. Choosing $i < j$ ensures that $(\forall i)(\lim_s f_s(a_i))$ exists) and by extensions of embeddings, f is a monomorphism. We have thus proven:

Theorem 2.2 *If $\mathbf{A}$ is Σ_1^- -presented, then $\mathbf{A}$ is Δ_2^- -isomorphic to a Π_1 subordering of $\mathbb{Q}$.*

Relativizations and similar arguments give:

Theorem 2.3 (Feiner [66], essentially)

- (i) If $\mathbf{A}$ is Σ_n -presented then $\mathbf{A}$ is Δ_{n+1} -isomorphic to a Π_n subordering of $\mathbb{Q}$.
- (ii) If $\mathbf{A}$ is Π_n -presented then $\mathbf{A}$ is Δ_{n+1} -isomorphic to a Σ_n subordering of $\mathbb{Q}$.

And hence

- (iii) If $\mathbf{A}$ is Δ_n -presented then $\mathbf{A}$ is Δ_n -isomorphic to a Δ_n subordering of $\mathbb{Q}$.

We remark that Theorem 2.3 can be extended to all the computably ordinals using either a ‘worker argument’ (see Section 8) or some other scheme using the uniformities involved at limit stages.

Thus we can (more or less) restrict ourselves to studying suborderings of $\mathbb{Q}$ (certainly up to Turing degree).

Definition 2.2 Henceforth we shall call a linear ordering Π_n (Σ_n , Δ_n , etc.) if it is a Π_n (Σ_n , Δ_n , etc.) subordering of $\mathbb{Q}$.

Note 2.1 It is important that the reader note that a Σ_n -presented linear ordering is different from a Σ_n linear ordering. Indeed, a Σ_n -presented ordering is isomorphic to a Π_n linear ordering by Theorem 2.3.

Using a very similar argument to that used in Theorem 2.1 it is possible to show that a Σ_1 linear ordering is isomorphic to a computably linear ordering. By relativization this means that a Σ_n linear ordering is isomorphic to a Δ_n linear ordering. By the limit lemma, we know that if A is a Δ_2 subset of $\mathbb{Q}$ then there is an approximation $g(\cdot, s)$ such that $g(x, s) \neq g(x, s+1)$ only finitely often, and with $\lim_s g(x, s) = A(x)$. Using this fact, again we can use a modification of Theorem 2.1 to show that a Δ_n linear ordering is isomorphic to a Π_1 linear ordering. By relativization it follows that a Δ_{n+1} linear ordering is isomorphic to a Π_n linear ordering, and hence a Σ_{n+1} linear ordering is isomorphic to a Π_n linear ordering. We now give an alternative proof of this result.

Theorem 2.4 (Feiner [66]) If $\mathbf{A}$ is a Σ_{n+1} linear ordering, then $\mathbf{A}$ is isomorphic to a Π_n linear ordering.

Proof. (Rosenstein [208, p. 431]) Recursively partition $\mathbb{Q}$ into half open intervals $\{I(r) : r \in \mathbb{Q}\}$ so that if $r <_{\mathbb{Q}} s$ then $I(r)$ is all left of $I(s)$, and so that given q we can compute the unique r , if any, with $q \in I(r)$. We can suppose there is a computable function p such that $\{p(x, r) : x \in \mathbb{N}\}$ enumerates $I(r)$. Now let A be a Σ_{n+1} linear ordering. It is not difficult to show there is a Π_n predicate R such that $r \in A \leftrightarrow (\exists x)R(x, r)$ and furthermore if $r \in A$ there is a unique x with $R(x, r)$. Define B via $q \in B \leftrightarrow (q = p(x, r) \text{ and } R(x, r))$. Then B is Π_n and evidently $B \cong A$. $\square$

It follows that the arithmetical hierarchy of order types for subordering of $\mathbb{Q}$ looks like

$$\Delta_1 \subseteq \Pi_1 \subseteq \Pi_2 \subseteq \Pi_3 \cdots \quad (2.1)$$

and again continue through the computable ordinals.

We now show that all the inclusions of (2.1) are proper.

Theorem 2.5 (Feiner [66, 67]) *For all n , there exists a Π_{n+1} linear ordering not isomorphic to a Π_n linear ordering and a Π_1 linear ordering not isomorphic to a computable one.*

Proof. By theorem 2.3, it suffices to construct a Σ_1 -presented linear ordering not isomorphic to a computable one, and then relativise the result.

Given an ordering $\mathbf{A}$, define the $(n-)$ block relation as $B(n)$: there is a set of n elements $x_1, \dots, x_n$ with $x_1 < x_2 < \dots < x_n$ such that (x_i, x_{i+1}) is a *successivity* (i.e., $(\neg \exists y)(x_i < y < x_{i+1})$) and for no y is (y, x) or (x_n, y) a successivity.

Then it is easy to see that $B(A) = \{n : B(n) \text{ holds}\}$ is Σ_3^A . The relevant definition is

$$\begin{aligned} n \in B(A) \text{ iff } & (\exists x_1, \dots, x_n)(\forall y)(\exists z) \\ & ((x_1 < \dots < x_n) \text{ and } \bigwedge_{i=1}^{n-1} \neg(x_i < y < x_{i+1}) \text{ and} \\ & (y < x_1 \rightarrow y < z < x_1) \text{ and } (x_n < y \rightarrow x_n < z < y)). \end{aligned}$$

The reader should note that $B(A)$ is a classical invariant.

In particular if $\mathbf{A}$ is computable the $B(\mathbf{A})$ is Σ_3 but if $\mathbf{A}$ is Σ_1 -presented and hence Δ_2 -presented, then $B(\mathbf{A})$ is apparently only Σ_4 (since to assert (e.g.) $x_1 < x_2$ we need to say that $(\forall s)(x_1 <_s x_2)$ of $\mathbf{A}$ is Σ_1 -presented).

In view of Theorem 2.3, it suffices to show that, given any set A we can construct an A -computably linear ordering $\mathbf{A}$ with $B(\mathbf{A})$ a set that is Σ_3^A complete (i.e., of degree A'''). Let D be any Σ_3^A set. If D is Σ_3^A then for some A -computably relation R we have

$$n \in D \text{ iff } (\exists x)(\forall y)(\exists z)R(x, y, z, n).$$

The ordering $\mathbf{A}$ we construct will have order type

$$\zeta + n_1 + \zeta + n_2 + \zeta + n_3 + \cdots$$

where the reader will recall ζ denotes the order type of the integers $\mathbb{Z}$ and we will use the n_i to represent D via $B(\mathbf{A})$. Since we can relativise to arbitrary A , we might as well take $A = \emptyset$ so D is Σ_3 .

We begin the construction with ω copies of ζ and construct a computably ordering of order type $\omega \cdot \zeta$. The x -th of the ω copies of η is labelled x . Fix x . We regard this ζ as being split as

$$\omega^* + 1 + 2 + 3 + 4 + 5 + \cdots$$

During the construction, we will add new points to the left and right of some of the blocks 1, 2, 3, ... above. Below we will have a notion of n being infinitively often “verified”. This verification machinery will select certain n to be ‘built around’. *Without* this machinery, the idea is simple. At stage s , put a point on each side the block we are building around each original n block (as well as at the end of the ω^* sequence). So at stage s we will have made the n block into a $s + n + s$ block.

So at stage 1 we’d put



At stage 2 we’d get:



Thus, *without* the verification machinery, at the end we’d get a sequence of ω blocks of type $\zeta = \omega^* + \omega$.

Now at stage s of the construction, we say (n, x) is verified (for y) if

- (i) for all $z < y$, (n, x) is verified for z ,
- (ii) (n, x) is not yet verified for y ,
- (iii) $R(x, y, t, n)$ holds for some $t < s$.

At stage s , for each n , if (n, x) becomes verified for some y , we define (n, x) as now verified and then in the x -th copy of ζ we also add a point *immediately* to the left and to the right of n (i.e., the original distinguished n -block). So, for instance, if $(2, x)$ were verified for all y but neither $(3, x)$ nor $(1, x)$ were, then at the end of the construction we'd have the x -th copy of ζ looking like:

$$\omega^* + \omega + 2 + \omega^* + \omega + \dots$$

as the 1 and 3 would be absorbed into the $\omega^* + \omega$'s. Now as $n \in D \leftrightarrow (\exists x)(\forall y)(\forall z) R(x, y, z, n)$ then we see that the only n which survive are those in S (namely those for which there is an x such that (n, x) is verified infinitely often). Hence $B(\mathbf{A}) = D$.

To finish the proof we need only choose some $A \leq_T \emptyset'$ with $\Sigma_3^A \not\leq_T \emptyset'$. This is a standard result from classical computability theory. Indeed for any A there is a Σ_3^A -complete S (so that $D \equiv_T A'''$). $\square$

Remarks 2.1

- (i) Feiner [66, 67] also proved that there were Σ_1 -presented *boolean algebras* not isomorphic to computably ones. This a much harder argument involving simultaneously coding $\mathbf{0}^{(n)}$ for all n . Since every computable boolean algebra can be represented as the boolean algebra of left closed right open intervals of a computable linear ordering, this result implies the linear ordering one. These ideas have been pushed further by Remmel, Goncharov, Thurber, Jockusch and Soare and others. We refer the reader to Downey [51] for further details.
- (ii) Define a set to be *low* if $A' \equiv_T \emptyset'$. It is known that there are low non-computably sets. The argument above shows that there are Π_1 linear orderings not isomorphic to low linear orderings. To a certain extent this is a drawback of the coding techniques used, since it is not clear if there are low linear orderings not isomorphic to a computably one. There are, as we shall see later.

- (iii) Feiner's techniques were quite novel at the time since nobody had ever coded anything except computably enumerable non-computably sets previously. The first significant simplification and systemisation of Feiner's theorem was in John Thurber's Thesis [237] and also can be found in Downey [51].
- (iv) For those interested in classical computability theory, the above results allowed Feiner to:
 - (a) show that the lattice of computably enumerable sets $\mathcal{E}$ is not computably presentable (using Lachlan's results on boolean algebras and hh-simple sets) and
 - (b) solve the strong homogeneity conjecture. This asked if, in the language of degrees with jump, whether the cones above any two degrees were isomorphic. The answer is no, using the above and initial segment results. (See Feiner [66, 67].) (Nowadays we use distributive lattices to get this, see Lerman [139].)
- (v) It is not difficult to completely characterise the computably presentable well orderings: indeed if A is any arithmetical subset of $\mathbb{Q}$ that is well ordered, then the order type of A is a computably ordinal and hence computably presentable. (Rice [200] and see Rosenstein [208, Theorem 16.37]. For related results we refer the reader to Pinus [184], Hay, Manaster and Rosenstein [92], Watnick [242] and Rice [200].)

One might wonder as to what complexity we might give an *order type*, rather than a specific *presentation*. One suggestion of Jockusch was: if $\mathbf{A}$ is a structure then the *degree of the isomorphism type* of $\mathbf{A}$ is the minimum of the degrees of all the presentations of $\mathbf{A}$. Unfortunately this is not a useful definition for linear orderings.

Theorem 2.6 (Richter [201]) *If $\mathbf{A}$ is an order type with a degree, then that degree is $\mathbf{0}$. In fact, if A is a given subordering of $\mathbb{Q}$, there is a $B \cong A$ such that the degrees of A and B have infimum $\mathbf{0}$.*

Proof. The argument is an easy diagonalisation of a type called the finite extension technique. Suppose A is a linear ordering not isomorphic to a

computably one. We build $B \cong A$ with B a subordering of $\mathbb{Q}$ via an isomorphism $f : B \rightarrow A$ in stages such that the degree of B and the degree of A has infimum $\mathbf{0}$. To do this we need to meet the requirements (for $e \in \mathbb{N}$):

$$R_{\langle e, i \rangle} : \Phi_e(B) = \Phi_i(A) = g \Rightarrow g \text{ is computable.}$$

At stage $s + 1$ we attend to R_s . Let $s = \langle e, i \rangle$.

Construction

STAGE 0:

Define $b_0 = 0$ and $f(0) = a_0$ (where $A = \{a_0, a_1, \dots\}$) and $n(0) = 0$.

STAGE $s + 1$:

Suppose we've defined $b_0, \dots, b_{n(s)}$ and $f(b_i) = a_i$ for all $i \leq n(s)$ so that this isomorphism from $\{b_0, \dots, b_{n(s)}\} = B_s$ to $A_s = \{a_0, \dots, a_{n(s)}\}$. Furthermore for all i and j , if (b_i, b_j) is a successivity in B_s , then either $|(f(b_i), f(b_j))| = \infty$ (in A) or $(f(b_i), f(b_j))$ is a successivity (in A).

STEP 1: Regard B_s as a string by identifying it with its characteristic function. Define a string μ to be *acceptable* if, whenever $\mu(k) = 1$, and $k \in (b_i, b_j)$ (in $\mathbb{Q}$) then $(f(b_i), f(b_j))$ is not a successivity in A (so that $(f(b_i), f(b_j))$ is infinite in A). Find the least string γ , if any extending B_s such that one of the following holds:

- (i) there is a z such that for all acceptable τ extending γ , $\Phi_s(\tau; z) \uparrow$
- (ii) for some z we have $\Phi_e(\gamma; z) \downarrow$ and $\neq \Phi_i(A; z)$.

If τ does not exist then $\Phi_i(A)$ would be computable, since to compute $\Phi_i(A; z)$ for any z , we need only find some acceptable $\rho \supset B_s$ such that $\Phi_e(\rho; z) \downarrow$ (one must exist). Then we know $\Phi_s(\rho; z) = \Phi_i(A; z)$. (We will only use acceptable extensions of B_s at each s .)

Our action is to set $B_{s+1}^* = \gamma$ if γ exists, and $B_{s+1}^* = B_s$ otherwise. That is, for each x with $\gamma(x) = 1$ put x into B_{s+1}^* .

STEP 2: Now suppose $B_{s+1}^* = \{b_0, \dots, b_m\}$. For those $j > n(s)$ with $j < m$ we extend f in the obvious way. We put b_j into the correct position in B_{s+1}^* determined by $\mathbb{Q}$ and find a_k in A to map these b_j to. This is possible as the string γ is acceptable. Now for each interval (b_i, b_j) in B_{s+1}^* with $(f(b_i), f(b_j))$ finite (in A) add new elements to make B_{s+1} with the property that if $(b_i, b_j) \in B_{s+1}$ then $|f(b_i), f(b_j)| = 2$ or ∞ . End of construction.

Evidently the construction succeeds since nothing we do at stage $t > s$ injures any action we did for R_s . Thus we diagonalise all the $R\langle e, i \rangle$ and hence $\deg(B) \cap \deg(A) = \mathbf{0}$. Clearly we have made f an isomorphism. $\square$

Actually the argument above holds for quite a wide class of structures and only relies on extensions of embeddings. Richter proved a general structure theoretical result which has the linear ordering result as a particular case.

Definition 2.3 (Richter [201, 203]) We define the *computable embedding condition* as follows. Given a structure A , a finite structure B and an embedding $f : B \hookrightarrow A$, define the class $A_{C,f}$ to be

$$\{D : D \text{ is a finite structure} \\ \text{extending } C \text{ embeddable into } A \text{ via a map extending } f\}.$$

Then A satisfies the computable extension property iff for all structures C isomorphic to a finite substructure of A , and for all functions f embedding C into A , the class $A_{C,f}$ is infinite and computable.

Theorem 2.7 (Richter [201, 203]) *For any countable structure A satisfying the computable extension property, there is a countable structure C isomorphic to A such that $\deg A \cap \deg C = \mathbf{0}$.*

The proof of Theorem 2.7 is similar to that of the Theorem 2.6. Furthermore, one can show without too much difficulty that Theorem 2.6 is a consequence of Theorem 2.7.

We remark that some structures can have degrees $\neq \mathbf{0}$, for example groups and lattices (see Richter [201]). We will look at the results there in more detail in Section 8.

The most fundamental question concerning presentations of linear orderings is

Question 2.1 (Open) *Classify the order types ρ such that ρ contains a computably member.*

Of course in Question 2.1 we'd like to have a classical classification. Certain things are obvious. For instance if $\mathbf{A}$ and $\mathbf{B}$ are computably presentable so is $\mathbf{A} + \mathbf{B}$, $\mathbf{A} \cdot \mathbf{B}$, etc.. It might appear from the above that we'd need to use n -blocks to make bad isomorphism types. Even here the answer is no.

Definition 2.4 Call a linear ordering A discrete if A contains no limit points.

It follows from a result of Watnick that there are Π_1 discrete linear orderings not isomorphic to computably ones. Watnick's theorem is:

Theorem 2.8 (Watnick [243], Downey [48]) A is a Π_{n+2} linear ordering iff ζA is a Π_n linear ordering.

Watnick's theorem has an interesting history in the sense that it has been rediscovered several times in different contexts (e.g., Downey [48], Ash-Jockusch-Knight [9]). To get a discrete Π_1 linear ordering not isomorphic to a computably one, one takes Watnick's theorem and applies it to a Π_3 linear ordering A not isomorphic to a Π_2 one. Then ζA is a discrete Π_1 linear ordering and not isomorphic to a computably one.

We will look at the proof of Watnick's theorem in Section 4.

Along similar lines, Downey and Knight proved a "1-jump" version of Theorem 2.8.

Definition 2.5 Let A be a linear ordering. Define $\gamma(A) = (\eta + 2 + \eta)A$.

Theorem 2.9 (Downey and Knight [55]) A is a Π_{n+1} linear ordering iff $\gamma(A)$ is a Π_n linear ordering. Furthermore, A is $\mathbf{a}'$ -presentable iff $\gamma(A)$ is $\mathbf{a}$ -presentable.

Proof. Let $n = 0$ and then relativise. Let A be a Π_1 linear ordering so that A is a Π_1 subset of $\mathbb{Q}$ (identified with $\mathbb{N}$ via a Π_1 subset of $\mathbb{Q}$ using Gödel numbers). Hence $x \in A$ iff $(\forall s)(R(x, s))$. Begin with a copy of $B = (\eta + 2 + \eta)\eta$. We can identify the x -th member of $\mathbb{Q}$ (i.e., under Gödel numbering) with a canonical successivity in $(\eta + 2 + \eta)\eta$ is the obvious way. Now perform a construction of "densifying" the ordering B by making the 2 in the x -th position become dense at stage s if $\neg R(x, s)$ holds. So the only remaining points at the end will correspond to the $x \in A$. $\square$

Corollary 2.10 There exists a Π_1 linear ordering A not isomorphic to a computably one with A having order type $(\eta + 2 + \eta)\tau$ for some τ . In particular, A has no n -blocks for $n \neq 2, 1$ and no ω or ω^* intervals.

We remark that Downey and Moses [59] have proved that every low discrete linear ordering is isomorphic to a computable one.

One quite natural approach is to look at increased levels of effectivity. An important notion is that of a *decidable* structure. This is one whose whole diagram is computable. That is, given any sentence we can effectively compute its validity in the structure. Along similar lines one can define a structure to be n -computable if we can decide effectively an n quantifier sentences. It is not hard to construct a computable ordering that is not 1-computable. We warm up with such a result.

Theorem 2.11 (Folklore) *There is a computable linear ordering A of order type ω with $S(x)$, the successor function, not computable. Hence A is not 1-computable since the adjacency relation is not computable.*

Proof. We give this result as a simple introduction to the priority method, and for this reason we don't give the simplest proof.

We shall build our computable order $A = \cup_s A_s$ in stages. We label the points $\{a_0, a_1, \dots\}$ for convenience. To make A have order type ω it suffices to satisfy, for each $e \in \mathbb{N}$, the requirement

$$N_e : a_e \text{ has } < \infty \text{ many predecessors.}$$

To make sure that $S(x)$ is not computable in A , we diagonalise over all possible algorithms for $S(x)$. To do this we satisfy for each $e \in \mathbb{N}$

$$P_e : \varphi_e \text{ is not the successor function on } A.$$

Here $\{\varphi_e\}_{e \in \omega}$ lists all p.r. functions. To meet P_e the basic idea is this. We pick a *follower* $x = x(e)$ and wait for a stage s where $\varphi_{e,s}(x) \downarrow$. If this never happens, then we win as φ_e is not total. If $\varphi_{e,s}(x) \downarrow$ we make sure that $\varphi_e(x)$ is not the successor of x by placing, if necessary, a new point $(s+1)$ between x and $\varphi_e(x)$.

This action gives potential conflict with N_j since if $\varphi_e(x) = a_j$ it may be that we put a new point to the left of a_j . If infinitely many P_e *together* were allowed to do this to a *single* a_j then we would fail to satisfy $N_j : a_j$ would have infinitely many predecessors.

The idea then is to have a priority ranking

$$N_0, P_0, N_1, P_1, \dots$$

of the requirements. We then only allow P_i to “injure” N_j if $i < j$. Here ‘injury’ means ‘add a point to the left of a_j ’. If we ensure that the total action we perform for each P_i within $i < j$ only injures N_j finitely often, then N_j will be happy since after a certain stage s_0 no new predecessors of a_j will be added. Note that we can’t predict s_0 but when this *finite injury priority method* is used one argues by simultaneous induction that all the P_j ‘require attention’ only finitely often and therefore all the N_e are injured only finitely often and are (hence) met.

The device that ensures we can do this is called the *coherence strategy* and the action we perform for a single requirement is called the *basic module*. We have described the basic module above. The coherence strategy here is very simple: we only pick followers x of P_k ($k \geq j$) if x is not a predecessor of a_j . Then P_k will never add a predecessor to a_j (if $k \geq j$).

The formal construction looks as follows. We say P_e *requires attention* at stage $s + 1$ if e is the least such that one of the following options holds.

- (i) P_e has no follower.
- (ii) P_e has a follower x and $\varphi_{e,s}(x) \downarrow$ and $\varphi_{e,s}(x)$ is the successor of x .

Construction

STAGE 0:

Set $A_0 = \{a_0\}$.

STAGE $s + 1$:

We are given A_s as $a_0, \dots, a_s$ in some specified order. Find the $e \leq s$ with P_e requiring attention (there will be one).

If (i) pertains it will be via some $x = a_{i_s}$ with $\varphi_{e,s}(x) = S(x) = a_{(i+1)_s}$ for some $a_{(i+1)_s}$. Put a_{s+1} between a_{i_s} and $a_{(i+1)_s}$ (killing $\varphi_{e,s}$ forever). Go to stage $s + 2$.

If (ii) pertains, put a_{s+1} to the right of A_s (i.e., $a_i < a_{s+1}$ for all $i \leq s$), and declare a_{s+1} to be a follower for P_e . End of construction.

It is not difficult to see that each P_e requires attention at most twice, and all the P_e and N_e are met so that A has the desired properties. $\square$

We spent a long time with the easy result above in the hope that the reader will understand the more elaborate uses of the machinery (and its extensions) later. In the future we will only sketch such proofs by specifying the basic modules and the coherence strategies.

Sometimes the concepts of n -computability and decidability are more closely related than the obvious fact that A is decidable iff it is uniformly n -computable for all $n \in \mathbb{N}$. For instance, the old result of Langford [134], shows the following. Recall that L is called discrete iff $L \cong \eta \cdot \hat{L}$ for some $\hat{L}$.

Theorem 2.12 (Langford [134]) *A discrete linear ordering is decidable iff it is 1-computable.*

The proof of Langford's Theorem uses quantifier elimination. We illustrate this technique in the next result which completely classifies 1-computability for computable linear orderings.

Theorem 2.13 (Moses [167, 168]) *A computable linear ordering is 1-computable iff the collection of adjacencies of L is a computable set.*

Sketch Proof. First one shows that if φ is a quantifier free sentence on variables $x_1, \dots, x_n$ then φ is equivalent to a disjunction of formulae of the form $\nu_1 \wedge \dots \wedge \nu_n$ where each ν_i is of the form $x_{\pi(i)} < x_{\pi(i+1)}$ or $x_{\pi(i)} = x_{\pi(i+1)}$ for some permutation π of $\{1, \dots, n\}$. We call this the standard form of φ . It follows that an existential formula γ is equivalent to a disjunction of the form

$$\exists \bar{x} \psi_1(\bar{x}, \bar{y}) \vee \dots \vee \exists \bar{x} \psi_m(\bar{x}, \bar{y}),$$

where each ψ_i is in standard form. Thus to see if γ is satisfiable, we need only look at each $\exists \bar{x} \psi_i(\bar{x}, \bar{y}) = \gamma_i$, say.

It is easy to see that each γ_i is equivalent to a finite set of statements of the form “ $y_i = y_j$ ”, “ $y_i < y_j$ ”, “there are at least k ($\leq n$) elements between y_i and y_j ”, “there are at least k elements $> y_i$ ”, and “there are at least k elements $< y_i$ ”. These are decidable iff the collection of successivities is computable. $\square$

Moses [unpubl.] has a classification hierarchy akin to the above for general orderings. We remark that Moses [171] has constructed for each n an n -computable ordering not isomorphic to an $n+1$ -computable one. Chisholm and Moses have also proven the following.

Theorem 2.14 (Chisholm and Moses [32]) *There exists a linear ordering $\widehat{L}$ which is n -computable for each $n \in \mathbb{N}$ but with $\widehat{L}$ not isomorphic to a decidable linear ordering.*

Here are some related questions.

Questions 2.2

- (i) *Are there n computable boolean algebras not isomorphic to $n+1$ computable ones for each $n \in \mathbb{N}$? What about a boolean algebra which is n computable for each $n \in \mathbb{N}$ but is not isomorphic to a decidable one?*
- (ii) *For each $n \in \mathbb{N}$, is there a finitely presented group which is n -computable but not $n+1$ computable? Can we construct a nondecidable finitely presented group which is n -computable for each $n \in \mathbb{N}$?*

Sketch Proof of Theorem 2.14. We sketch the proof of Theorem 2.14 particularly since it illustrates Ehrenfeucht-Fraïssé games. The argument is a finite injury priority argument. For structures P and Q we will write $P \equiv_{\Sigma_n^0} Q$ to mean that P and Q satisfy the same Σ_n^0 sentences. The idea is to build a sequence

$$L_0, L_1, L_2, \dots$$

of decidable linear orderings with limit $\widehat{L}$ so that for each n and each $m > n$, we ensure that for all $\bar{a} \in \widehat{L}$ we can effectively find a $\bar{b} \in L_m$ such that $\langle \widehat{L}, \bar{a} \rangle \equiv_{\Sigma_n^0} \langle L_m, \bar{b} \rangle$ and conversely. This will ensure that $\widehat{L}$ is n -computable by the decidability of L_n and playing an Ehrenfeucht-Fraïssé game between $\widehat{L}$ and L_n .

To ensure that $\widehat{L}$ has no decidable copy, we will ensure that L_n is one of L_n^+ or L_n^- . (As usual, we are building all the orderings in stages, and at each stage they are finite.) We begin by building $L_n = L_n^-$ and only change when we see an opportunity to diagonalise. We will have a specific sentence φ_n used to distinguish between L_n^+ and L_n^- , so $L_n^+ \models \varphi_n$ and $L_n^- \not\models \varphi_n$. We ensure that the theory of $\widehat{L}$ is not computable and hence $\widehat{L}$ cannot have a computable copy. Thus the process and the switching is simply controlled by an opponent trying to enumerate the theory of $\widehat{L}$ and we wait till the opponent enumerates $\neg\varphi_n$ to switch.

Let $T_1, \dots, T_k$ be orderings. Then

$$\sigma(T_1, \dots, T_k)$$

denotes the *shuffle sum* of $T_1, \dots, T_k$ which is defined by taking a copy of $\mathbb{Q}$ and densely putting copies of each of the T_i 's in. (So between T_i and T_j there is always a copy of T_q for any i, j, q .)

Now $L_0 = \sigma(1, 2, 3)$. For sequence $1, \dots, k$, let $1, \dots, \widehat{j}, \dots, k$ denote the subsequence obtained by leaving out j . We define $L_1^0 = \sigma(1, 2, 3)$, and define $L_1^1 = \sigma(2, 3)$, $L_1^2 = \sigma(1, 3)$, and $L_1^3 = \sigma(1, 2)$. Finally define

$$L_1^- = \sigma(L_1^0, L_1^1, L_1^2) \quad \text{and} \quad L_1^+ = \sigma(L_1^0, L_1^1, L_1^2, L_1^3).$$

More generally, we will know that

$$L_{n-1} = \sigma(T_1, \dots, T_k).$$

We will let $L_n^0 = L_{n-1}$, and

$$L_n^j = \sigma(T_1, \dots, \widehat{T_j}, \dots, T_k).$$

Then, as above,

$$L_n^+ = \sigma(L_n^0, \dots, L_n^{k-1}) \quad \text{and} \quad L_n^- = \sigma(L_n^0, \dots, L_n^k).$$

Notice that in a stage by stage construction we can switch from L_n^- to L_n^+ easily. Also note that each L_n^j misses an L_{n-1}^p for some p . (This is what the T_i 's are.) However, each L_{n+1}^w has all the L_{n+1}^p 's appearing in arbitrary order. One now proves by induction that each L_n^j with $n > 1$ satisfies a certain Π_{n+1} sentence φ_n^j that is not satisfied by any L_n^k for $k \neq j$ nor any shuffle of them. (For instance, φ_1^j says that there is no maximal block of size j .) Second, one proves by induction that all the L_n^p are $\equiv_{\Sigma_{n-1}^0}$ to each other or any shuffle of each other. The result will then follow by the diagonalisation of the construction, and these two facts. $\square$

Corollary 2.15 (Chisholm and Moses [32]) *There is a structure that is intrinsically n -computable for each $n \in \mathbb{N}$ (i.e., each computable structure isomorphic to it is n -computable) yet is not isomorphic to a decidable structure.*

Proof. Take the ordering $\widehat{L}$ above and add constants c_i for each member of the ordering. $\square$

3 Computable Isomorphisms

A natural suggestion would be to look at ‘effective’ order types instead of classical order types. That is, since we are, after all, looking at effective procedures on computable orderings, surely the ‘correct’ classification would be to consider those computable linear orderings computably isomorphic to a given one. Does this make a difference? As we shall see in this (and later) sections, it certainly does. We have already seen that computable properties are not, in general, invariant. For instance, the natural presentation of $\omega = \{0 < 1 < \dots\}$ has the property that the successor function $S(x)$ is computable. However in Theorem 2.11 we observed that there is a computable copy of ω with $S(x)$ not computable. It follows that there are two isomorphic computable linear orderings that are not *computably isomorphic* (as $S(x)$ being (non) computable would be preserved by a computable isomorphism). Consideration of this example leads to the following basic definition.

Definition 3.1 We call a computable linear ordering A *computably categorical* (sometimes *autostable* in the literature) if any computable linear ordering B isomorphic to A ($B \cong A$) is computably isomorphic to A ($B \cong_{\text{comp}} A$).

Clearly all finite linear orderings are computably categorical. Some infinite ones are too.

Example 3.1 (Folklore) η is *computably categorical*.

To see this, we effectivise Cantor’s back-and-forth argument. Let $A = \{a_0, a_1, \dots\}$, $B = \{b_0, b_1, \dots\}$ be two dense computable orderings without end points. We define $f_0(a_0) = b_0$. At stage $2s$ we ensure that $a_s \in \text{dom } f$. To do this we will have defined $\text{dom } f_s = a_0, \dots, a_{s-1}, a_{i_0}, \dots, a_{i_{s-1}}$ in some order with f_s a partial isomorphism. Now assuming $a_s \notin \text{dom } f_s$, we see where a_s lies relative to the points in $\text{dom } f_s$. If, for instance, $a_i < a_s < a_j$ we simply use the density of B to find some b_k with $f_s(a_i) < b_k < f_s(a_j)$ and define $f_{s+1}(a_s) = b_k$. At odd stages $2s + 1$ we similarly ensure that $b_s \in \text{ran } f$ by using the density of A . By extensions of embeddings, $f : A \cong_{\text{comp}} B$. $\square$

The computably categorical linear orderings were classified by Remmel.

Theorem 3.1 (Remmel [193]) *A computable linear ordering $\mathbf{A}$ is computably categorical iff the (classical) order type of $\mathbf{A}$ has only a finite number of successivities.*

Proof. If $\mathbf{A}$ is a computable linear ordering with only finitely many successivities then $\mathbf{A}$ has order type $(\sum_{i=1}^q n_i + \eta) + n_{q+1}$ with $n_i < \infty$ for all i . To show that $\mathbf{A}$ is computably categorical, suppose $\mathbf{B} \cong \mathbf{A}$ with $\mathbf{B}$ computable. then $\mathbf{B}$ must also have the same type and hence if we match the n_i in $\mathbf{B}$ to those in $\mathbf{A}$ we can extend to a computable isomorphism by the technique of Example 3.1.

Conversely, suppose $\mathbf{A}$ has infinitely many successivities. We build $\mathbf{B} \cong \mathbf{A}$ with $\mathbf{B} \not\equiv_{\text{comp}} \mathbf{A}$. To do this, we use a priority argument and meet the requirements ($e \in \mathbb{N}$):

$$N : \mathbf{B} \cong \mathbf{A}$$

$$P_e : \varphi_e \text{ is not an isomorphism from } \mathbf{B} \text{ to } \mathbf{A}.$$

We will build $\mathbf{B}$ in stages as $\mathbf{B} = \cup_s B_s$ and will ensure that $f : \mathbf{B} \rightarrow \mathbf{A}$ is a (Δ_2-) isomorphism by ensuring that we meet (for all $e \in \mathbb{N}$):

$$N_e : \lim_s f_s(b_e) \text{ exists}$$

(i.e., we have $f_{s+1}(b_e) \neq f_s(b_e)$ only finitely often),

$$M_e : f(b_e) \text{ is defined (i.e., } f \text{ is a function)}$$

$$\widehat{M}_e : (\exists y)(f(y) = a_e) \text{ (i.e., } f \text{ is onto).}$$

Additionally we ensure that f is injective. The key to the construction is the basic model for P_e .

The idea is as follows. If we ever see φ_e not 1-1 (or if for some x , $\varphi_e(x_i) \uparrow$, although we can't know this) we need not worry about P_e . Assuming these don't happen, we really need to do something to meet P_e . At stage s we will have ensured that f_s is a partial isomorphism taking $B_s = \{b_0, \dots, b_s, b_{i_0}, \dots, b_{i_s}\}$ to $\{a_0, \dots, a_s, a_{i_0}, \dots, a_{i_s}\} = A_s$. (This meets M_e and $\widehat{M}_e$ provided we meet N_e .)

Suppose in A_s we *knew* that (a_i, a_j) was a successivity. Let $f_s(b_k) = a_i$ and $f_s(b_m) = a_j$, say. For the sake of P_e we need do nothing until we see a stage $t > s$ such that for some b_n, b_r we have

$$(i) \varphi_e^t(b_n) \downarrow \text{ and } \varphi_e^t(b_r) \downarrow$$

$$(ii) \varphi_e(b_n) = a_i \text{ and } \varphi_e(b_r) = a_j$$

$$(iii) (b_n, b_r) \text{ is a successivity in } B_t.$$

Now we can force φ_e to be wrong at stage $t+1$ by putting a new element b_g between b_n and b_r , since then (b_n, b_r) is not a successivity in B yet (a_i, a_j) is a successivity in A .

Obviously, if we are to keep $f = \lim_s f_s$ an isomorphism, the strategy above will entail us redefining f_{t+1} . For instance, we may have $f_{t+1}(b_n) = a_u$ and $f_{t+1}(b_r) = a_v$. When we put b_g between b_n and b_r unless a new element appears between a_u and a_v , we'd need either to define $f_{t+1}(b_g) = a_u$ or $f_{t+1}(b_g) = a_v$ and hence either make $f_{t+1}(b_n) \neq f_t(b_n)$ or $f_{t+1}(b_r) \neq f_t(b_r)$ (as we need f injective). Indeed, it is possibly the case that we will need to reset $f_{t+1}(b)$ either for all $b \in B_{t+1}$ with $b \geq b_r$ or for all $b \in B_{t-1}$ with $b \leq b_n$.

Here is where we use a priority argument. We do not allow this action for P_e to take place if it injures N_j for $j < e$. That is, we don't let P_e make $f_{t+1}(b_j) \neq f_t(b_j)$ for any $j < e$. Now this causes the following problems. Suppose that we have points b_i and b_j both with $i, j < e$ and it happens that $b_i \leq b_n < b_r \leq b_j$. Now it may be that we have defined both $f_t(b_i)$ and $f_t(b_j)$. Since $i, j < e$ they have higher priority than P_e and so *we cannot add a new point b_g between $f_t(b_i)$ and $f_t(b_j)$ at any stage $\geq t$* . Of course, *if such a new point appears at some stage $t_1 > t$* we are free to pursue the strategy of the basic module and still keep $f_{t_1+1}(b_i) = f_t(b_i)$ and $f_{t_1+1}(b_j) = f_t(b_j)$. That is, *only change f_{t_1} on the open interval (b_i, b_j)* .

The trouble is that we *cannot know* if another point will appear, but we now have an obvious strategy: while we *wait* for some such new point to appear we begin a *new strategy on a new successivity* in A . If the interval $f(b_i), f(b_j)$ is finite, after a finite number of such tries we will get a pair of points (b_{i_1}, b_{i_2}) *outside* of the *finite* interval (b_i, b_j) . If the interval is *infinite*, then eventually we will see a new point appear and can *return* to the old strategy. In this way we can make the N_j and P_e strategies cohere since we will get to meet P_e on *some* pair.

The final problem not addressed in the above is the following: we may not be able to computably identify the successivities in A . Being a successivity is only Π_1 and it is not clear that A may be isomorphic to a computable linear ordering with computable successivities. Indeed this is not always the case:

Theorem 3.2 (Rommel, Goncharov, after Feiner) *There is a computable linear ordering A such that if B is computable and $B \cong A$ then B does not have computable successivities.*

Proof. (Downey) One way to get Theorem 3.2 is to use the following. Take a Π_1 linear ordering C not isomorphic to a computable one. Now apply the

Downey-Knight [55] construction 2.5 to get $A = (\eta + 2 + \eta)C$ computably presentable. Then the order type of successivities of A are isomorphic to C and if they were computably presentable so would C be, which it is not. $\square$

Proof of Theorem 3.1 concluded. To overcome this final problem, we guess if (a_i, a_j) is a successivity. If it is not, as we will find out, we abandon it and try another pair (a_{i_1}, a_{j_1}) . The trick is to list all the pairs (a_i, a_j) by Gödel numbers, and let P_e always use the *least* pair. Clearly we will eventually settle on a real successivity. To complete the proof one simply formalises the above, and the result will follow by a standard application of the finite injury technique. This concludes our sketch of the proof of 3.1. $\square$

It is not difficult to interweave other ‘finitary’ requirements with the above argument. For example, if $L_1, L_2, \dots$ is any computable list of isomorphic linear orderings one can diagonalise against each of them simultaneously.

Corollary 3.3 (Rummel [193])

- (i) *A classical order type of a computable linear ordering contains either one, or infinitely many computable order types.*
- (ii) *If L is a computable linear ordering with infinitely many successivities then there is no computable list $L_1, L_2, \dots$ of all computable linear orderings isomorphic to L .*

Corollary 3.3 seems an appropriate place to mention another recurrent theme in computable combinatorics: *the effective content of classical theorems*. We will look at this theme in more detail in the sections to follow, but will confine ourselves here to some brief comments on Corollary 3.3.

A famous and deep combinatorial theorem is Laver’s theorem [136] on the better-quasi-ordering of countable linear orderings under embeddability. One form of this result is:

Theorem 3.4 (Laver [136]) *Let $\rho_1, \rho_2, \dots$ be a countable collection of countable order types. Then there is some $i < j$ with ρ_i embeddable into ρ_j .*

Corollary 3.3 says that the obvious computable analogue of Theorem 3.4 fails for computable order type. That is, if $\rho_1, \rho_2, \dots$ are computable order types they may all be pairwise *computably* nonembeddable. (Although

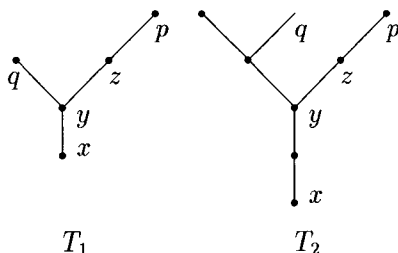
Corollary 3.3 (i) only speaks of nonisomorphism, the proof of Theorem 3.1 actually gives computable nonembeddability.)

Since we know some ρ_1 embeds into ρ_j one might ask exactly how complicated such an embedding would be. For example, must there be an arithmetical embedding? Richard Shore [221] has the decisive material here. He has shown that it is possible to choose the collection of order types so that there is not even a hyperarithmetical witnessing function.

At this point I cannot help but mention some beautiful recent developments related to Theorem 3.4. One of the great achievements of the twentieth century is Gödel's famous incompleteness theorem which roughly states that in any sufficiently strong system with an computably enumerable set of axioms, there is a formula *statable within* the system, true of the system, but not provable *within* the system. Unfortunately, whilst Gödel's theorem has a major impact on the foundations of mathematics, many 'working' mathematicians regarded it as irrelevant to their studies, since the formulae found by Gödel were 'not interesting' and were 'things one would not wish to prove'.

First Paris and Harrington [182] and then Harvey Friedman and others (see Simpson [223]) destroyed this thesis by producing 'unprovable "mathematical" theorems' in a number of systems. We will confine ourselves to a single example.

Let T_1 and T_2 be finite trees regarded as lattices. We say T_1 embeds into T_2 if there is an injective function $f : T_1 \rightarrow T_2$ preserving meets (i.e., $f(x \wedge y) = f(x) \wedge f(y)$). See the example in the diagram below:



Kruskal [121] proved that if $T_1, T_2, \dots$ is an infinite collection of finite trees, then there is some $i < j$ with T_i embeddable into T_j . This statement is not provable in 'finite combinatorics' (i.e., Peano arithmetic (PA)) but neither is it *statable* in PA. Friedman found the following finite form of Kruskal's theorem:

Theorem 3.5 (H. Friedman) *Let n be given. There exists a number $k = f(n)$ so large that if $T_1, \dots, T_k$ are finite trees with $|T_i| \leq n \cdot i$, then there exists $i < j$ with T_i embeddable into T_j .*

Whilst Theorem 3.5 is clearly statable in PA, Friedman proved that 3.5 cannot be *proven* in PA and any proof of 3.5 necessarily involves *infinite* sets. (Indeed Friedman found variations of 3.5 that need *uncountable* sets.) The way to prove this is to show that the witness function $n \rightarrow f(n)$ grows so fast as to not be provable in PA. The reader is referred to Simpson [223] and Friedman-Simpson-Smith [76] for further results. These logical investigations have had some fine consequences in classical combinatorics. For example, Friedman's result above was a crucial ingredient of the Robinson-Seymour theorem that the set of all finite graphs is well-quasi-ordered under the relations ' H is minor of G ', and as a consequence there is a 'Kuratowski' theorem for surfaces.

Returning to our story, Remmel's theorem 3.3 has been generalised in several ways. First we could ask the more general question of classifying when a computable linear ordering is Δ_n -categorical for some n . That is, try to classify (by order type ρ) when, given any two computable linear orderings A and B of type ρ there is a Δ_n -isomorphism from A to B . Similarly we might ask when *every* isomorphism is Δ_n . This latter property is called Δ_n -stability. For instance it is easy to see that ω is Δ_2 -stable and (hence) Δ_2 -categorical.

This leads to the powerful recent techniques developed by Chris Ash and by others. We will look at these in Section 8, when we will be able to sketch the technical machinery needed for Ash's results.

A second method of generalisation is to look at linear orderings as merely an example of a more general class of structures. This approach was taken by Ash-Nerode [13], Goncharov [83], Barker [15], Harizanov [88, 89, 90] and others using *Model Theory*. Since we are trying to keep this account relatively 'nonlogical', again we give only a bare outline. A model is a structure

$$(A, R_1, \dots, R_n, \dots, F_1, \dots, F_n, \dots, C_1, \dots, C_n, \dots)$$

where A is the universe, the R_i are relations on A , the F_i are functions of A and the C_i are distinguished constants. For example, a group is a structure of the form $G = (G, \cdot, 1)$. In the language of the structure we can write sentences and study all of the structures obeying those sentences. The class

of models of the sentences

$$(\forall x, y, a) (x \cdot (y \cdot z) = (x \cdot y) \cdot z)$$

$$(\forall x, y) (x \cdot y = y \cdot x)$$

$$(\forall x) (\exists y) (x \cdot y = y \cdot x = 1)$$

$$(\forall x) (x \cdot 1 = x)$$

is simply the set of abelian groups and a model of these sentences is an Abelian group. A model is computable if $A = \mathbb{N}$ and the R_i , F_i and C_i are a computable collection of computable relations, functions and constants respectively. Dzgoev and Goncharov [62] discovered a condition called *branching* and showed that if A is a computable model which branches, then Q is not computably categorical. This result has applications including Remmel's theorem and similar results classifying the computably categorical boolean algebras (La Roche, Goncharov) and p -groups (Smith). The only problem with Dzgoev and Goncharov's (and other) results is that it is often quite difficult to verify that a given structure branches. (Sometimes this is much more difficult than a direct proof.) One open question here is:

Question 3.1 (Open) *Develop a generalisation of branching for Δ_n -categoricity.*

A very attractive result here is due to Moses. If R is a relation on a computable linear ordering $\mathbf{A}$, we call R *intrinsically computable* if in every computable linear ordering $\mathbf{B}$ classically isomorphic to $\mathbf{A}$ via $F: A \cong B$ we have $f(R)$ computable. (The notion is due to Ash-Nerode [13].)

Theorem 3.6 (Moses [167]) *If $\mathbf{A}$ is a computable linear ordering, then a computable relation R on $\mathbf{A}$ is intrinsically computable iff R is equivalent in $\mathbf{A}$ to a quantifier free formula $\theta(x_1, \dots, x_n, a_1, \dots, a_n)$ with $a_1, \dots, a_n$ in $\mathbf{A}$.*

The final generalisation of Remmel's theorem we will look at is to study the 'fine structure' of isomorphism types of computable orderings and see what can be preserved. In the remaining sections we see several of these types of results. For this sections we will only look at the situation where we

equip the linear orderings with additional computable properties. First we know that not all linear orderings have copies with computable successivities. Remmel [188, 191, 192] has a number of results in this area as corollaries to his results on boolean algebras. These investigations are related since as we mentioned earlier any computable boolean algebra can be represented as the boolean algebra of the left closed right open subintervals of a computable linear ordering (similarly Δ_n , Σ_n , Π_n). Here is a quick proof.

Theorem 3.7 (The Representation Theorem, folklore after Stone) *Every countable boolean algebra is isomorphic to a interval algebra $\text{Intalg}(L)$ of a linear ordering of the same degree.*

Proof. This is just an effective version of Stone's theorem. We represent a given B as an algebra of sets. Thus let $B = \cup_s B_s$ with $B_0 = \{0, 1\}$, and $B_{s+1} - B_s = \{b_s\}$. We will define at each stage a subalgebra $\widehat{B}_s$ containing B_s .

Define L_0 the ordering with two points labelled 0 and 1. (Then $\text{Intalg}(L_0)$ consists of the two sets $\emptyset$ and $[0, 1)$.) So we have the induced mapping g_0 with $0 \mapsto \emptyset$ and $1 \mapsto [0, 1)$. At stage $s + 1$ we will have a set of $\text{Atoms}_s = \{a_{s_1}, \dots, a_{s_n}\}$ listing the atoms of the subalgebra of $\widehat{B}_s$ together with the linear ordering $L_s = 0, x_{s_1}, \dots, x_{s_n} = 0$ so that $g_s(a_{s_j}) \mapsto [x_{s_{j-1}}, x_{s_j})$ inducing an isomorphism from the subalgebras $\widehat{B}_s$ to $\text{Intalg}(L_s)$.

At stage $s + 1$ if b_s is in $\widehat{B}_s$ we need to do nothing. Otherwise, for each $a = a_{s_i}$ such that b_s splits a (i.e., both $x \wedge b_s$ and $x \wedge \overline{b_s}$ are nontrivial), add a new point y to L_{s+1} between $x_{s_{j-1}}$ and x_{s_j} to split the interval $[x_{s_{j-1}}, x_{s_j})$ into $[x_{s_{j-1}}, y) \cup [y, x_{s_j})$. Naturally we map $a_{s_j} \wedge b_s$ to one of them, say, $[x_{s_{j-1}}, y)$ and map $a_{s_j} \wedge \overline{b_s}$ to $[y, x_{s_j})$. Note that this generates two new atoms for $\widehat{B}_{s+1}$, also of $\text{Intalg}(L_{s+1})$. Let $c_1, \dots, c_m$ denote the atoms of $\widehat{B}_{s+1}$ below b_s . Clearly we have ensured that the induced map $g(b_s) = g(c_1) \cup \dots \cup g(c_m)$ works. The result follows. $\square$

Notice that in the above, atoms correspond to successivities, so when Remmel studied computable boolean algebras with computable atoms he had corollaries about computable linear orderings with computable successivities. Unfortunately there is no direct transfer the other way, since many nonisomorphic linear orderings give rise to the same boolean algebra.

Question 3.2 (Open) *Determine the relationship between the degrees of linear orderings and the boolean algebras they generate.*

If a linear ordering has computable successivities, then we can sharpen Corollary 3.3.

Theorem 3.8 (Remmel [193]) *If A is a computable linear ordering with an infinite computable set of successivities, then*

- (i) *There is a computable collection $B_1, B_2, \dots$ of computable linear orderings, each isomorphic to A , such that if $i \neq j$ then the sets of successivities of B_i and B_j are Turing incomparable.*
- (ii) *We can arrange that the sets of successivities S_i of each B_i are immune (that is, there is no infinite computably enumerable subset of S_i).*

As we will see in Section 6, Theorem 3.8 (i) does *not* hold if we drop the computable successivities' hypothesis. The proofs of Theorem 3.8 (i) and (ii) (and similar results) come from finite injury arguments blended with the techniques used in Corollary 3.3. See Remmel [193] for details.

Another variation comes from restricting the sorts of structures we look at. A nice question here was asked by Nerode, Remmel and others: classify the order types of computably categorical linear orderings with computable successivities. That is, we want order types ρ such that if A and B have types ρ and both have computable successivities then $A \cong_{\text{comp}} B$. This question proved quite hard and was finally solved by Moses.

Theorem 3.9 (Moses [167, 168]) *The computably categorical linear orderings with computable successivities (in the sense above) are precisely those with order types $\sum_{i=1}^n (k_i + g_i) + k_{n+1}$ where k_i is finite and $g_i \in \{\omega, \omega^*, \zeta\} \cup \{d \cdot \eta : d \in \mathbb{N}\}$ for all i .*

Moses' proof of Theorem 3.9 is not obvious and involves a surprising intermediate step. Recall from the last section that a linear ordering is called n -computable if the Σ_n formulae uniformly denote computable relations. Note that ' (x, y) is not a successivity' is an example of a Σ_1 relation so to say A has computable successivities is to assert that a *particular* Σ_1 -relation on A is computable. However, we recall from section 2, Theorem 2.13:

A computable linear ordering $\mathbf{A}$ is 1-computable iff $\mathbf{A}$ has computable successivities.

Define a linear ordering to be *good* if it has the order type described in Theorem 3.9. We need the following definition.

Say that $\mathbf{B}$ is *weakly good* if $\mathbf{B}$ has order type $(\sum_{i=1}^n k_i + g_i) + k_{n+1}$ with each k_i finite and $g_i \in F_n$ for some $n = n(i)$, with F_n denoting the class of linear orderings with no block of size $> n$.

Definition 3.2

- (i) The *block relation* $B(x, y)$ is defined to hold if there are only finitely many elements between x and y .
- (ii) The relation $M_k(x)$ holds if the block containing x has exactly k elements.

Moses proved Theorem 3.9 using Theorem 2.13 and the following sequence of results.

Theorem 3.10 (Moses [167, 168])

- (i) If $\mathbf{A}$ is a 1-computable linear ordering with the block relation computable, then the order type of Z is weakly good iff every 1-computable linear ordering isomorphic to A has the block relation computable.
- (ii) If $\mathbf{A}$ is a 1-computable linear ordering with $M_i(x)$ for $i \in \mathbb{N}$ all computable, then A has order type $(\sum_{i=1}^n k_i + g_i) + k_{n+1}$ with each k_i finite and $g_i \in \{q \cdot \eta : q \in \mathbb{N}\}$ iff every 1-computable linear ordering has $M_i(x)$ for $i \in \mathbb{N}$ computable.
- (iii) If $\mathbf{A}$ is a computably categorical 1-computable linear ordering, then it has $B(x, y)$ computable.
- (iv) If $\mathbf{A}$ is a computable linear ordering with no blocks of infinite length and $B(x, y)$ computable, then the isomorphism type of $\mathbf{A}$ contains a 1-computable linear ordering with $B(x, y)$, the predecessor relation $P(x)$ computable and the successor relation $S(x)$ computable.

To finish the proof, Moses assembles the facts above. First suppose that $\mathbf{A}$ is of good order type. Then a standard back and forth argument like that used in Corollary 3.3 shows that $\mathbf{A}$ is computably categorical for 1-computable linear orderings, then by Theorem 3.10 (iii), $\mathbf{A}$ has computable block relation and hence $\mathbf{A}$ is weakly good by Theorem 3.10 (i). We can then partition $\mathbf{A}$ by a finite number of points $P_0 < \dots < P_m$ with each interval $(-\infty, P_0)$, (P_0, P_1) , $\dots$, (P_m, ∞) of order type in $F_{n(i)} \cup \{\omega, \omega^*, \zeta\}$.

Let $\mathbf{B}$ be the result of deleting from $\mathbf{A}$ all orderings of type ω , ω^* , or ζ . Then $\mathbf{B}$ is computably categorical, 1-computable and has by 3.10 (iv) $B(x, y)$, $S(x)$ and $P(x)$ all computable, as $\mathbf{B}$ has no infinite blocks. This makes $M_i(x)$ computable for all i and hence by 3.10 (ii) $\mathbf{B}$ has order type $(\sum_{i=1}^n k_i + g_i) + k_{n+1}$ with k_i finite and $g_i \in \{q \cdot \eta : q \in \mathbb{N}\}$. It follows that the order type of $\mathbf{A}$ is good, and we are done. $\square$

We should note that the proof of Moses' theorem is greatly complicated by the analogous result that complicated Remmel's theorem 3.3. Recall that this was that there are computable linear orderings $\mathbf{A}$ with no recursive linear orderings in the order type of $\mathbf{A}$ having computable successivities. Here Remmel [193] used a coding argument (i.e., like Feiner's argument) to construct a 1-computable linear ordering $\mathbf{A}$ with the property that if $\mathbf{B}$ is a computable linear ordering isomorphic to $\mathbf{A}$ then the block relation is not computable in $\mathbf{B}$. Moses and Hingston (see Moses [167]) used another coding argument to construct a 1-computable linear ordering not isomorphic to a 2-computable linear ordering. The above result classifies the computably categorical 1-computable linear orderings. Naturally the following question suggests itself.

Question 3.3 (Open) *Classify the computably categorical η -computable linear orderings (i.e., amongst n -computable linear orderings).*

Schwartz [219] investigated the computable categoricity amongst linear orderings with the block relation computable. He used a finite injury priority argument to show:

Theorem 3.11 (Schwartz [219]) *The linear orderings with the block relation computable that are computably categorical in this class are exactly those of order type*

$$f_0 + k_0 + f_1 + k_1 + \cdots + f_n + k_n + f_{n+1}$$

where each f_i is finite and $k_i \in \bigcup_{q \in \mathbb{N}} q\eta$.

Proof. We follow Schwartz's thesis [219]. (Actually there is a slight flaw in the proof in Schwartz's thesis which is not difficult to repair, but Schwartz has not published this.) A routine back and forth argument shows that the above order types are sufficient. Conversely, suppose $\mathbf{A}$ is an order type not

of the above form. Then we must build $B \cong A$ such that B has a computable block relation and $B \not\cong_{\text{rec}} A$. So we must ensure that

$$R_e : \varphi_e(B) \neq A \text{ if } \varphi_e \text{ is the } e\text{-th computable partial isomorphism.}$$

There are two cases:

CASE 1:

For some $a \in A$, the block of a is infinite. So a lies in a block of type ω , ω^* or $\omega^* + \omega$. In this case it is not difficult to see that we can build $B \cong A$ by using the identity outside of a 's block and the Remmel strategy on a 's block to kill all the φ_e . Note that the block relation is computable as a 's is.

CASE 2:

Otherwise. Each block in A is finite. It is not hard to see that as A is infinite the order type of A is $\sum \{h(q) : q \in \mathbb{Q}\} + h(+\infty)$ where h is a function from $\mathbb{Q} \cup \{-\infty, \infty\}$ to $\mathbb{N}$.

The strategy is to diagonalise against all the φ_e . The basic idea is to play φ_e on blocks. That is, for two points x and y in B , ensure:

$$|[x, y]|_{(\text{in } B)} \neq |[\varphi_e(x), \varphi_e(y)]|_{(\text{in } A)}.$$

There are two subcases.

CASE 2a:

$ra h$ is unbounded. The strategy here is quite simple. In A wait till we see two points a and b in the same block and wait till $\varphi_e(x) = a$ and $\varphi_e(y) = b$ for some x and y . Now we can use a Remmel strategy. Wait until we see some block $[a_1, b_1]$ occur in A with more that $|[a, b]|$ many elements so that we can also add elements and move $[x, y]$ to map to $[a_1, b_1]$. We may need to do this several times as we do not know $|[a, b]|$. Actually, this argument can be simplified by the following result of Moses.

Theorem 3.12 (Moses [167]) *If A is a computable linear ordering with computable block relation, A is isomorphic to a 1-computable linear ordering.*

Sketch Proof. Let A be a computable linear ordering with a recursive block relation. Build 1-computable $B \cong A$. In view of Theorem 2.13 it suffices to construct B with computable successivities. Now at stage 0 we map a_0 to b_0 . Suppose at some stage s we see x in a_0 's block with $x < a_0$, say. Map b_s to x

and declare b_s to be b_0 's predecessor. If we see at some $t > s$ some z occur with $x < z_a < a_0$ we *change the maps* and set $b_s \rightarrow z$ and $b_t \rightarrow x$. Since x is in the same block as a_0 we'll need to do this at most finitely often, and the above can be assembled to give a Δ_2 isomorphism from B to A . $\square$

[*Technical aside re Theorem 3.12:* As we mentioned previously it is possible to construct a 1-computable A not isomorphic to any computable B with recursive block relation (Moses). Furthermore, although A is *isomorphic* to some 1-computable with recursive block relation, A itself may not have computable successivities. Of course if A is computably categorical it must be 1-computable.]

Proof of (3.12) continued. The result above clearly simplifies case 2a since we can assume A is a 1-computable linear ordering and hence know the size of the interval $[a, b]$. This fact similarly helps with case 2b below.

CASE 2b:

rah is bounded. Suppose $\text{rah} = \{n_0, \dots, n_k\}$, where these all occur infinitely often. Here we can take $n_0 < n_1 < \dots < n_k$. Since A is not of the correct order type, there must be either an ω or ω^* sequence of pairs of blocks (p_i, q_i) with $|p_i| < |q_i|$. Using this fact we can again use Rummel 'isomorphism changing' to start with a block we assume to be of size p_i in A and then kill φ_e by making its pre-image of size q_i . Using the usual priority methods means that we will settle down on the correct size. $\square$

We remark that Schwartz's result can be obtained by using Theorem 3.10 together with Theorem 3.12.

Another area some authors have examined is that of *discrete* linear orderings. The reader will recall that these are the linear orderings where each element has a successor and a predecessor. From section 2 the reader will recall that discrete linear orderings have special properties *since a discrete linear ordering is 1-computable iff it is uniformly n -recursive for all n and hence decidable*.

We shall further examine discrete linear orderings later, but for this section concern ourselves with only one set of results.

Let **D** denote the class of (order types of) decidable discrete linear orderings, **BD** the subclass of **D** with (additionally) the block relation computable and **SBD** the subclass of **BD** with the relation " x and y are in

adjacent blocks” computable. We can use Feiner-type coding arguments to show $\mathbf{D} \not\subseteq \mathbf{BD} \not\subseteq \mathbf{SBD}$. However, as we shall see, they have a much more precise relationship.

If $\mathbf{A}$ is a discrete linear ordering then $\mathbf{A} \cong \zeta\rho$ for some order type ρ . We call ρ the *finite condensation* of $\mathbf{A}$ (Rosenstein [208]) and write $C_F(\mathbf{A}) = \rho$. It is easy to see the following:

Observation 3.1 (Moses [170])

- (i) $\tau \in \mathbf{BD}$ iff $C_F(\tau)$ has a computable copy.
- (ii) $\tau \in \mathbf{SBD}$ iff $C_F(\tau)$ has a 1-computable copy.

The main result of Moses [170] is

Theorem 3.13 (Moses [170]) $\tau \in \mathbf{D}$ iff $C_F(\tau)$ is isomorphic to a Π_1 linear ordering.

Proof. We prove this in some detail, as we will need to modify it for the more difficult arguments to come. The nontrivial direction is to show that if $C_F(\tau)$ is Π_1 then $\tau \in \mathbf{D}$.

Let $\mathbf{A}$ be a Π_1 linear ordering with $x \in A$ iff $(\forall x)(R(x, s))$ and $A \leq \mathbb{Q}$. We use the *amalgamation of blocks* technique. We have $A = \cap_s A_s$ with each A_s computable. This follows since we can regard $x \in A_s$ iff $(\forall t < s)(R(x, t))$. Let $A_s = \{a_{0,s}, a_{1,s}, \dots\}$ in order of Gödel numbers.

We begin the construction at stage 0 by putting a point z on each side of $a_{i,s}$ for all $i \in \omega$. This defines a block $B(a_{i,s}, 0) = z a_{i,s} z$. (To avoid confusion, we will call such new points z -points. An a_i -point can become a z -point but not conversely.) The rough idea is that at stage 1 we add another pair of z -points to each of these blocks to get (roughly) $B(x, 0) = z x z$ with $B(x, 1) = z z x z z$ and so forth to give $\omega + \omega^*$ around x .

The only trouble at some stage s is if x dies. That is we see $\neg R(x, s)$ hold. Now we want to *stop* building around x and get rid of what we have done. We can’t exactly do this, but we can incorporate our work for x into an adjacent block. (Pick the closest and if there is a choice pick e.g., the right one.) So

$$z z z x z z z \quad z z z x' z z z$$

$$x\text{-block} \quad x'\text{-block}$$

at stage 3 perhaps becomes the amalgamated block

$$\begin{array}{c} zzzz zzzz \quad zzz x' zzzz \\ x'\text{-block} \end{array}$$

at stage 4. Note that x has become a ‘ z ’ point.

To see that the resulting ordering is decidable, it suffices by Langford’s result to see that the successivity function is computable. But this is easy. For a pair x, y present at stage n (x, y) is a successivity at stage $n + 1$, iff (x, y) is a successivity. Evidently the order type that results is correct, giving the proof. $\square$

4 Theorems of Watnick and of Lerman

In this section we shall introduce a quite fundamental proof technique: the infinite injury (Π_2 –) priority method. We will do so via a proof of a theorem of Watnick [242] which evolved from some earlier and simpler work of Lerman [138]. We will look at the exact context of Watnick’s and Lerman’s original results later in Section 5, but here we will be concerned with extending the Moses results of the end of Section 3 (e.g., Theorem 3.13) to computable orderings.

First we shall begin with some (possible obscure) general comments. In the finite injury priority method we must build an object C to satisfy certain conditions which are arranged in a priority list

$$L_0, L_1, L_2, \dots$$

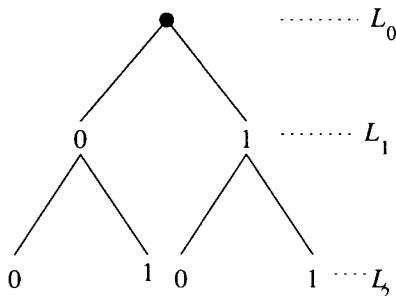
The problem is that we don’t know if we ever need to act for the sake of L_i and that an L_i -action can injure all L_j -actions for $j > i$. However if we can arrange matters so that each L_i only acts finitely often the L_j will eventually reach an environment where it can be met since it will be protected by the priority orderings.

In a typical infinite injury argument, the situation is more complex. Now we have a priority list

$$L_0, L_1, \dots$$

where some L_i ’s can act *infinitely often*. Hence at no stage in the construction can we afford to pretend L_i has *finished acting*. Usually we will have several

versions of L_j , for $j > i$, guessing L_i 's behaviour. We can also do this for a finite injury argument as follows. Here, for simplicity, suppose that if L_i is not injured, it will act at most once. The strategy for the L_i can be thought of as a binary tree:



We interpret 1 as ' L_i does not act' and 0 as ' L_i acts'. Thus the string $1^{\frown}1^{\frown}0$ encodes ' L_0 and L_1 don't act' and ' L_2 acts'. In the course of the construction we have a path δ_s with $\lim_s |\delta_s| \rightarrow \infty$ and δ_s moving left. For instance, if the construction appears as $1^{\frown}1^{\frown}0^{\frown}1$ at stage 3 and then L_1 acts, the correct next string would be $1^{\frown}0^{\frown}1^{\frown}1$. Note that we have 1 not 0 in the L_2 position since L_2 has not acted in a manner consistent with the guess $1^{\frown}0$ (i.e., after L_1 acts). This corresponds to L_2 being 'injured' by L_1 acting. The finite injury method is often nowadays called the Π_1 -method since we only need a Π_1 -oracle to figure out exactly how we meet the L_i , that is, to figure out the *true path* (TP) of the construction on the tree. This is defined as the left most path visited infinitely often: if $\sigma \subset \text{TP}$ then $\sigma^{\frown}0 \subset \text{TP}$ iff $(\exists^\infty s)(\sigma_s \subset \text{TP})$, otherwise $\sigma^{\frown}1 \subset \text{TP}$.

In the Π_2 method, the requirements are such that (using the tree model above) the path δ_s can move both left and right. We must ensure that the *versions* of the L_i on the tree path have the correct environment to be met. Plainly this only works with certain requirements. The keys are to get

- (i) the 'wrong guesses' not to interfere too much, and
- (ii) an environment where the 'correct guesses' all cohere.

The following example will hopefully clarify these vague points. It is not difficult to see that if γ has a computable copy then $C_F(\gamma)$ has a Π_2 copy (i.e., is a sub-ordering of $(\mathbb{Q}, <)$) by counting quantifiers (compare with

Theorem 3.13). Rosenstein [208] asked if the reverse implication held, i.e., if $C_F(\gamma)$ has a Π_2 copy does γ have a computable copy? The following is due to Watnick [243] but was independently and subsequently rediscovered by Downey [48], and Ash, Jockusch and Knight [9]. (See also Roy and Watnick [212].)

Theorem 4.1 (Watnick [243]) *If γ discrete then $C_F(\gamma)$ has a Π_2 copy implies γ has a computable copy. That is, ρ is Π_2 presentable iff $\zeta\rho$ is computably presentable.*

Proof. The proof uses the ‘amalgamating blocks’ strategy of Theorem 3.13, but uses Π_2 -guessing to control how we amalgamate blocks. The proof is thus an infinite injury argument. Before we give the formal construction, we shall give some motivational remarks to indicate how our construction works.

Let $A \subseteq (\infty, <)$ be a Π_2 -copy of $C_F(\gamma)$. Without loss of generality, we may suppose A is infinite. We let $\mathbb{Q} = \{x_1, x_2, \dots\}$ be a computable enumeration of $\mathbb{Q}$.

A nice representation of A . First we take a nice representation of A as follows. By standard representation of a Π_2 set, we may suppose without loss of generality that $A = \{f(i) : \varphi_{f(i)} \text{ is total}\}$ where $\{\varphi_i : i \in \omega\}$ lists the partial computable functions. We replace this representation by a better one (given by a tree which controls strategies). Define a stage s to be a σ -stage (for $\sigma \in 2^{<\omega}$) as follows by induction of $\ell h(\sigma)$ (the length of σ):

- (i) Every stage s is a $\emptyset$ -stage.
- (ii) If s is a τ -stage and $\ell h(\tau) = j = f(i)$.

then if $\varphi_{f(i)}^s(y) \downarrow$ where $y = \mu z$ ($z \notin \text{dom } \varphi_{f(i)}^t(y) : t \text{ is a } \tau\text{-stage and } t < s$) we say s is a $\tau\hat{0}$ -stage. Otherwise s is a $\tau\hat{1}$ -stage.

This gives us our best representation of A . Namely, A is the ‘true path’ of the above tree: let β be the true path, that is, β is the leftmost path visited infinitely often so that $\emptyset \subset \beta$ and if $\sigma \subset \beta$ we have $\sigma\hat{0} \subset \beta$ iff $\exists^\infty s$ (s is a $\sigma\hat{0}$ -stage), otherwise $\sigma\hat{1} \subset \beta$. Then

$$A = \{j : \ell h(\sigma) = j \text{ and } \sigma\hat{0} \subset \beta\}.$$

One nice property of this representation is that if $j_1, \dots, j_n$ is any finite subset of A then $j_1, \dots, j_n$ all ‘appear to be in A ’ together infinitely often. The above representation trick was first noticed by Jockusch [101].

Intuitively, we say that x_j *appears to be in* A at stage s if s is a σ -stage and $\ell h(\sigma) = j$. We define $\sigma_s =$ the unique string with $\ell h(\sigma_s) = s$ and s is a σ_s -stage. Then we say that A *appears to be* $\{j : \tau \hat{\cap} 0 \subseteq \sigma_s \text{ and } \ell h(\tau) = j\}$ at stage s , and formally define A_s to be this. For simplicity, we shall also ask that $\forall s (x_0 \in A_s)$.

A good model for $(B, <) = (C_F^{-1}(A), \leq)$. We must perform three basic tasks:

- (i) Around those points $x_i \in A$ we must build $\omega^* x_i \omega$,
- (ii) we must ‘incorporate’ all those ‘bad’ $x_j \notin A$ somehow into such blocks,
- (iii) we must not, ourselves, build anything else.

A good model for B is given as follows: at stage s we have a set of balls with various markings on them, arranged in a line. We have a supply of new balls we must add to this line, either inserted or added to the ends. These new balls will be z -balls, y -balls, or x_i -balls. The intention is that z -balls are attempting to be part of an ω^* -block, y -balls part of an ω -block and x_i -balls part of A . Later we may change our minds and convert y to z or x_i to y or z . *However, if an x_i -ball turns into a y -ball it can't change back.*

The line of balls we refer to as the ‘surface’. An x_i ball on the surface will be marked with a *guess* $\sigma \in 2^{<\omega}$ where $\ell h(\sigma) \geq i + 1$. If there is a stage s where this guess ‘proves wrong’ we turn this x_i ball in to a y or a z ball (with no guess).

The x_i -balls we must place on the surface at stage s are simply those that appear to be in A at stage s . Roughly speaking, we must place the y ’s and z ’s around the x_i ’s that appear in A at s .

The strategies. To satisfy our three aims we must have a reasonably complicated strategy dictating where to place our new balls at each stage. This strategy must overcome several problems whose solutions we outline below:

Incorporations. As a first approximation let us suppose that we have three points which appear in A at stage s in the order

$$x_j < x_i < x_k.$$

Now suppose that *really* $x_j, x_k \in A$ and $x_i \notin A$. What we shall know is that x_j and x_k appear in A together infinitely often, but x_i only appears to be in

A finitely often. For simplicity let us suppose that x_j and x_k are successors in A . Thus $\forall p (x_j \leq p \leq x_k \text{ and } p \in A \rightarrow p = x_j \text{ or } p = x_k)$ we then see that locally $B = C_F^{-1}(A)$ should be one copy of $\omega^* + \omega$ at x_j and $\omega^* + \omega$ at x_k with x_i not there. To achieve this, we shall incorporate x_i into x_j 's block. (The entire construction is 'left justified'). We do this as follows. At stage s , when x_j appears to be in stage s , we must add one z before x_j and one y following x_j . As it stands, we put z immediately before x_j but put y as far right as possible to be consistent with our current picture of A at stage s .

That is to place y for the sake of x_j we go as far right as possible until we see an x_p also appearing in A (or get to the end of A_s) we then repeat the process to x_p .

For example a typical situation might be

$$\begin{array}{l} zzz x_j yz x_i yz x_k \cdots x_p \cdots \quad \text{at stage } s-1 \\ zzzz x_j yz x_i yz x_k \cdots x_p yz x_p \cdots \quad \text{at stage } s \\ \underbrace{\hspace{10em}}_{\text{no change}} \end{array}$$

Here, at stage s it appears that $x_j \in A_s$ and x_p is the next 'apparent member' of A_s . Now this idea will work since it will ensure that since x_j and x_k appear in A together infinitely often we build infinitely many z 's before x_j and similarly before x_k . Also, since x_i appears in A only finitely often we build only finitely many y 's and z 's between x_j and x_k and thereafter almost always incorporate x_i in x_j 's ω -block whenever x_j and x_k appear in A together. Thus this builds

$$\omega^* x_j \underbrace{yy \cdots yzz \cdots z x_i yyy \cdots y}_{\text{finite}} \underbrace{\cdots y \cdots \omega^* x_k}_{\omega} \cdots$$

and so $\omega^* x_j \omega \omega^* x_k$ as required. We call this strategy 'incorporation' as in Theorem 3.13, since we try to ensure that if x_i doesn't appear in A infinitely often, then x_i gets incorporated into somebody's block.

The problem. Thus far we have discussed how we attempted to place the y - and z -balls. The crucial property that will allow this strategy to work is that between any two successive points of A (e.g., like (x_j, x_k)) we must ensure that the wrongly placed 'false points' like x_i can be turned into an ω -ordering (i.e., incorporated into our blocking).

Now perhaps infinitely often there will be new x_t occurring where perhaps $x_j < x_t < x_k$ in the $\mathbb{Q}$ -order and x_j, x_t, x_k all appear in A when x_t appears

in A , say at stage s (x_t) = s_t . Since, at such a stage s_t it appears that $x_j, x_t, x_k \in A$, it seems reasonable that we should put x_t between x_j and x_k . The crucial idea now is how we can relate such x_t to x_i .

From some point on it will never seem that $z_i \in A$ so it seems unreasonable that we should put anything between x_j and x_i . That is, although in the $\mathbb{Q}$ -order perhaps $x_j < x_t < x_i$, if it doesn't seem that $x_i \in A$ at a stage when we must place an x_t -ball on the surface, we shall place x_i beyond x_t in the following order $x_j < x_i < x_t < x_k$ (as x_k appears in A at the same time).

Thus assuming we are working to the right of x_0 , our guiding principle is that we always try to put new x_i 's as far right as we can.

This brings us to a minor point. When a new x_i appears on the scene, we first determine if $x_i < x_0$ or $x_0 < x_i$. If $x_i < x_0$ we work as above but everything goes left. *Thus, without loss, we shall assume that x_0 is the least member of A , and only work right of x_0 .* This simplifies presentation.

Summarising so far, our key idea is that we don't build between any x_p and x_q if x_q is x_p 's current successor and x_p doesn't appear to be in A at stage x . This idea helps us to overcome the problems induced by our strategy above.

The next situation we must consider is the situation when x_t, x_j, x_i, x_k above are *all in* A but infinitely often it appears that $x_j \in A, x_i \notin A$ and $x_k \in A$, and infinitely often it appears that $x_j \in A, x_i \notin A$ and $x_t \notin A$.

Following our strategy above, perhaps at stage s_0 when x_t appeared for the first time we set down the balls in the order

$$x_j \ x_i \ x_t \ x_k \quad (\text{at stage } s_0)$$

when their 'real order' is $x_j \ x_t \ x_i \ x_k$. We do so since it appears that $x_i \notin A$ at stage s . But later we see that $x_i \in A$ and we see also x_j, x_t and $x_k \in A$ at some stage $s_1 > s_0$. We now realise that our initial guess as to x_t 's position appears wrong and now put

$$x_j \ x_t \ x_i \ \hat{x}_t \ x_k.$$

Here $\hat{x}_t$ denotes a group of balls around x_t 's 'old position'. Now we can't remove these balls but also don't wish to build an $\omega^* + \omega$ block around $\hat{x}_t$, so our solution is to relabel the $\hat{x}_t$ -balls as y - or z -balls.

This then gives rise to another problem. Later we again see $x_t \in A$ but now $x_i \notin A$. We can't use the x_t position between x_j and x_i since we really only put it there in the first place since x_j and x_i appeared in A . Perhaps

in another triple $x_i \notin A$ and $x_t \in A$ but infinitely many such x_t get inserted between x_j and x_i . We really should put new x_t as far right as we possibly can consistent with our current picture of A_s . If, later, it appears that x_j and x_i are in A , we again build around the x_t we put between x_j and x_i last time and cancel its current position. This leads to the fundamental idea of the section.

Labelling. Whenever we put an x_t on the surface into a new position for the first time we give it the label $\sigma \subseteq \sigma_s$ with $lh(\sigma) = t + 1$. Thus we are saying: here is where x_t 's correct position is, should σ be correct. Now, if we must move x_t , we do so because some x_i encoded in this guess, which appeared to be *out* of A , now appears to be *in* A , and $i < t$. (Of course, in σ_t this will appear as $\tau \hat{\small{1}}$ with $lh(\tau) = i$.) The condition $i < t$ is simply to sort out which ball to move. When we move x_t , we give its new position a 'better' label. The correct position for x_t is a stable position *corresponding to the leftmost label visited infinitely often*. The reader should note that, at any particular time, x_t might have several positions labelled on the surface. Only one (at most) is correct. Here we need another notion. Let $\leq_L$ denote the lexicographic ordering on the tree. Thus $\sigma \leq_L \tau$ iff $\sigma \subseteq \tau$ or $(\exists \gamma)(\gamma \hat{\small{0}} \subseteq \sigma \text{ and } \gamma \hat{\small{1}} \subseteq \tau)$. The reader should read $\sigma \leq_L \tau$ as $\sigma = \tau$ or σ is *stronger* than τ . Note that only those $\sigma \leq_L \tau$ will always have their apparent positions uncanceled. Those $\tau \leq_L \sigma$ only get visited finitely often and so only move σ finitely often. Hence we shall argue that σ reaches a stable position and so does x_t . We now give the formal details of the construction, although we hope that readers can see them for themselves.

Construction, stage $s + 1$.

Construction of $C_F^{-1}(A) = B$, say stage $s + 1$.

STEP 1 (Cancellation): Compute σ_s . Cancel all positions marked τ for $\tau \not\leq_L \sigma_s$. Regard these now as y -balls and similarly any z -balls associated with them become y -balls.

STEP 2 (Placing x_j -balls): In order of j , for each τ with $lh(\tau) = j$ and $\tau \hat{\small{0}} \subseteq \sigma_s$, proceed as follows:

If there is currently an (uncanceled) position marked $\tau \hat{\small{0}}$ on the surface, do nothing. If there fails to be such a position, establish one by placing an x_j -ball marked $\tau \hat{\small{0}}$ as far right as possible. This will be on the right of B_{s-1} unless there is an x_i -ball marked $\gamma \hat{\small{0}} \subseteq \tau \hat{\small{0}}$ and $x_j < x_i$. In this case, for

the $<_{\mathbb{Q}}$ -least such x_i , we place x_j immediately left of x_i and put s z -balls immediately preceding x_i . ($x_j z z z \cdots z x_i$).

STEP 3 (Placing y -, z -balls): Now place y - and z -balls as indicated in the discussion. That is place another z -ball before x_0 . Now go right and find the first x_j -ball, if any, marked $\gamma^{\wedge} 0 \subseteq \sigma_s$ for some γ . If there are no y - or z -balls between x_j and its predecessor, add a y - followed by a z -ball. If there are, already, a block of y 's and a block of z 's preceding x_j , add one further y to the y -block and one further z to the z -block. Now continue until we get to the right end. Here, add one y -ball. End of construction.

Verification. For the sake of the following lemmata we shall adopt the following definitions:

Definition 4.1 We say a ball n appears to be in an x -ball g 's ω^* -block at stage s if n is a z -ball and, if m is any ball with m between n and g then m is a z -ball.

Definition 4.2 We say a ball n appears to be in an x -ball g 's ω -block at stages if g has guess σ for some $\sigma \subseteq \sigma_s$ (and g is not cancelled at s), $g \leq n$, and one of the following conditions is satisfied:

- (i) (a) there exists an x -ball g in $\text{dom} B_s$ with $g \leq n < \hat{g}$ such that $\hat{g}$ has guess $\subseteq \sigma_s$ such that
- (b) there does not exist an x -ball in $\text{dom} B_s$ with guess $\subseteq \sigma_s$ and $g < r \leq n$, and
- (c) there exists a y -ball p in $\text{dom} B_s$ such that $n \leq p < \hat{g}$, or
- (ii) there does not exist an x -ball $\hat{g}$ in $\text{dom} B_s$ with $g < \hat{g}$.

Remark 4.1 The intuition here is that either n occurs beyond the largest apparent member of A at s (in 4.2 (ii)) or n occurs before the z -block of two apparently consecutive (at s) elements of A at stages.

We also say that position g (an x -ball) receives attention at stage s if the number of elements in g 's apparent ω^* -block at s increases.

Let β denote the leftmost path. That is, as in the discussion, β is the leftmost path visited infinitely often.

Lemma 4.2 (Stable position lemma) *Every x_i reaches a stable position. That is, either $x_i \notin A$ in which case x_i receives attention finitely often (in total) or $x_i \in A$ in which case there is a unique x_i -ball that receives attention infinitely often (and is, of course, never cancelled). This ball is marked $\sigma \hat{0} \subseteq \beta$ with $lh(\sigma) = i$.*

Proof. By induction. Suppose for all $j < i$ the lemma holds. Let s_0 be a stage such that $\forall s_0 (\sigma \hat{0} \leq_i \sigma_s)$.

Let $s_1 > s_0$ be the least $\sigma \hat{0}$ -stage exceeding s_0 . Now at stage s_1 , choosing s_0 minimal, we may suppose that we place an x_i -ball h , marked $\sigma \hat{0}$. We claim that this is x_i 's stable position.

First, this position cannot be cancelled, we only cancel positions when their guess appears wrong (in step 1 of the construction). By Definition 4.2 (i) above this cannot occur. Hence this position is never cancelled.

Second, this position receives attention infinitely often since $\sigma \hat{0} \subset \beta$ and so $\exists^\infty s (\sigma \hat{0} \subseteq \sigma_s)$ by definition of β .

Finally h_i is unique. To see this let g be another x_i -ball that receives attention infinitely often. This ball g must have guess γ , for some γ with $lh(\gamma) = i + 1$. There are three possibilities: either $\gamma \leq_L \beta$ and $\gamma \not\subseteq \beta$, $\gamma \subset \beta$ or $\beta \leq_L \gamma$. In the first case, there are only finitely many γ -stages. We only add to this x_i -ball's z -block at γ -stages (by construction) and so such an x_i -ball can receive attention at most finitely often.

If $\gamma \subset \beta$ then $\gamma = \sigma \hat{0}$. There are two possibilities. Either g was appointed at a stage t before x_i (i.e., before s_1) or g was appointed after h , so at a stage $s_2 > s_1$. In the first case our assumptions concerning the minimality of s_1 mean that there is a η -stage $\hat{t}$ with $t \leq \hat{t} \leq s_0$, with $\eta \leq_L \gamma$ and $\eta \neq \gamma$. Such a stage cancels g . If g is appointed after stage s_1 then g was appointed at a $\sigma \hat{0}$ -stage. We only appoint new positions if there is not already a $\sigma \hat{0}$ -position available. There is, of course, namely the one occupied by h . Thus we wouldn't have appointed g after all.

Finally, if $\beta < \gamma$ then $\sigma \hat{0} \not\subseteq \gamma$. By step 1, at each $\sigma \hat{0}$ -stage we cancel any x -balls marked γ . Hence g gets cancelled and so wouldn't have received attention after all. $\square$

Lemma 4.3 (True z -ball lemma) *Let $x_i \in A$. Let $\hat{x}_i$ denote x_i 's stable position (given by Lemma 4.2). Suppose n is an ω^* -ball of $\hat{x}_i$ at stage s_1 . Then n is a z -ball of $\hat{x}_i$ at every $\sigma \hat{0}$ -stage, where $\sigma \hat{0} \subseteq \beta$ and $lh(\sigma) = i$.*

Also if $z(n, s)$ is the number of balls between n and $\hat{x}_i$ at stage s then for all $s \geq s_1$, $z(n, s) = z(n, s_1)$ ($= z(n)$, say).

Proof. Let $n, \hat{x}_i$ be as above at stage s_1 . By definition of ω^* -ball, n is a z -ball and there are no balls between n and $\hat{x}_i$ save for z -balls. The construction (in step 2) specifically ensures that when we place new x -balls $x_j < \hat{x}_i$ we do not disrupt any currently placed z -balls. In particular, no new γ_j balls can be placed between n and $\hat{x}_i$. This, of course, means that $z(n, s) = z(n, s_1) = z(n)$ since new z -balls are always placed on the *left* end of $\hat{x}_i$'s apparent ω^* -block. $\square$

Lemma 4.4 (True ω^* -block lemma) *Let $x_i \in A$ and $\hat{x}_i$ as in Lemma 4.3. Then in B there appears an ω^* $\hat{x}_i$ -block.*

Proof. By Lemma 4.3 and the fact that $\hat{x}_i$ receives attention infinitely often, and hence we add infinitely many z -balls before $\hat{x}_i$. $\square$

Definition 4.3 Suppose n is a z -ball of some $\hat{x}_i$. Then we say n is a *stable ω^* -ball*, and we say n *adheres to $\hat{x}_i$* (as an ω^* -ball).

Lemma 4.5 (ω -adherence lemma) *Let m be any ball and suppose m is not a stable x_i -ball or a stable ω^* -ball. Then there is a stable x_i -ball $\hat{x}_i$ and a stage $s(m)$ such that*

- (i) *m appears to be an ω -ball of x_i at each $\sigma \hat{0}$ -stage $> s(m)$ where $\sigma \hat{0} \subseteq \beta$ with $\ell h(\sigma) = i$.*
- (ii) *If $d(m, \hat{x}_i, s)$ denotes the number of balls between m and $\hat{x}_i$ at stage s then $\forall s > s(m)$ ($d(m, \hat{x}_i, s) = d(m, \hat{x}_i, s(m)) = d(m, \hat{x}_i)$).*

(In the case above, we say m adheres to x_i as an ω -ball.)

Proof. This is the main lemma that our machinery — discussed before the construction — is meant to achieve. Now whether m is an unstable z -ball, an unstable x_j -ball for some j or a y -ball. All of the cases are similar and can, roughly speaking, be treated simultaneously. Let s be the stage when m was placed on the surface. If m was an x_j -ball let $c(m) = m$. If m was a y -ball, find the $\leq$ -greatest x -ball x_j alive at stage s with $x_j \leq m$ and set $x(m) = x_j$. (Note: x_j *doesn't* need to be apparently in A at s . Also x_j

may later die or x_j may be $\hat{x}_j$.) Finally if m was a z -ball, find the $\leq$ -least x -ball x_j with $m \geq x_j$ and set $c(m) = x_j$. Note that m appears to be in x_j 's z -block at stage s . Since there are no dormant x -balls between m and $c(m)$, as in Lemma 4.3, we cannot place more balls between $c(m)$ and m after stage s by the way we place balls. It therefore suffices to argue the lemma for $x(m) = x_j$ instead of m .

Without loss, $x_j = c(m)$ is unstable, and either x_j is eventually cancelled or x_j is never cancelled but only appears in A finitely often.

Now let $\hat{x}_j$ be the $\leq$ -greatest stable position $\leq x_j$ in B at stage s .

Claim 4.1 *After stage s we can't add a stable position $\hat{x}_t$ with $\hat{x}_i < \hat{x}_t < x_j$.*

Suppose not, and $\hat{x}_t$ is such. Let $\sigma \hat{0}$ be the guess of x_t . The only time we place such a ball $\hat{x}_t$ between balls already on the surface is because the guess forces us to. That means there must be some ball x_q with guess $\gamma \hat{0}$, say and $\hat{x}_i < \hat{x}_t < x_q < x_j$ forcing $\hat{x}_t < x_j$. Since this is so, by prior ties of movement it must be that $q < t$.

For suppose otherwise, and $t < q$. Now x_q must be already present on the surface at the stage s when x_i enters. Since this is a *new* position for x_t there must be no position marked $\sigma \hat{0}$ on the surface at stage s . Hence it cannot be that the guess $\gamma \hat{0}$ of x_q extends $\sigma \hat{0}$, because if this was so we would already have put down a x_t -ball g marked $\sigma \hat{0}$ by the time we put $\gamma \hat{0}$ down for x_q . But then this ball g cannot have been cancelled in the intervening stages since if g were cancelled, so too x_q would have been cancelled. (g would be cancelled since $\sigma \hat{0} \not\leq_L \sigma_s$ but then as $\sigma \hat{0} \subseteq \gamma \hat{0}$, $\gamma \hat{0} \not\leq \sigma_s$.) Hence we see $q < t$.

Now as $q < t$ and $\hat{x}_t$ is a stable position we must have that x_q is also a stable position. (Remember, in this case $\sigma \hat{0} \subseteq \gamma \hat{0}$ and $\tau \hat{0} \subseteq \beta$. If x_q 's position is cancelled, so too is anything, in particular $\hat{x}_t$, marked $\sigma \hat{0}$.)

In either case we see that no such x_q (and hence $\hat{x}_t$) can exist.

Thus we have Claim 4.1 that there are no stable positions between $c(m) = x_j$ and $\hat{x}_i$. We now show that x_j adheres to $\hat{x}_i$.

Let $x_{i_1}, \dots, x_{i_n}$ list those x_0 -balls alive at stage s with $\hat{x}_i < x_{i_1} < \dots < x_{i_n} = x_j$. Let $s(m)$ be the least stage such that for all j with $1 \leq j \leq n$ we have

- (i) either x_{i_j} is cancelled at stage $s(m)$, or
- (ii) $\forall s > s(m)$ ($\gamma_j \not\leq \sigma_s$ where γ_j is the guess of x_{i_j}).

Such a stage must exist by Claim 4.1 and the definition of stability. As the notation suggests, we claim that this $\hat{x}_i(s(m))$ is correct, namely that

Claim 4.2

- (i) x_j appears to be an ω -ball of $\hat{x}_i$ at each σ^0 -stage $> s(m)$.
- (ii) $d(x_j, \hat{x}_i, s(m)) = d(x_j, \hat{x}_i, s)$ for all $s > s(m)$.

By construction, Claim 4.2 (ii) $\Rightarrow$ (i) because of the way we place y -balls and the fact that $\hat{x}_i$ is stable. We argue that Claim 4.2 (ii) holds in a similar way to our argument that there are no stable x -balls between x_j and $\hat{x}_i$. Suppose $d(x_j, \hat{x}_i, s(m)) < d(x_j, \hat{x}_i, s)$ for some least $s > s(m)$.

There are two ways we might insert new balls between x_j and $\hat{x}_i$. Either the new ball n is a y - or a z -ball placed between $\hat{x}_i$ and x_j because we see that x_q appears in A at s (i.e., its guess appears correct) for $\hat{x}_i < x_q < x_j$ (notice that $x_q < x_j$ and $q \neq i_k$ for any $1 \leq k \leq n$ by choice of $s(m)$) or the ball is a new x -ball x_d , say, placed there because again we see x_q whose guess appears correct with $\hat{x}_i < z_q < z_j$. Again $x_q \neq x_j$ or x_{i_k} for any $1 \leq k \leq n$. We claim that in either case no such z_q can exist.

Now, using the reasoning of Claim 4.1, x_q which did not exist at stage s must have appeared at a stage where some of the x_{i_k} for $i \leq n$ looked correct, since otherwise we'd placed it beyond x_j . Also using the same reasoning as Claim 4.1 it must be that q exceeds those x_{i_k} that forced it in (otherwise x_{i_k} would be cancelled first) and so since x_q 's guess *extends* such x_{i_k} 's guess, x_q 's guess appears correct at best only when x_{i_k} 's does too. But now, choice of $s(m)$ means that x_{i_k} 's guess never again looks correct. Thus x_q 's guess also never again looks correct. Hence stage s can't exist after all. This clinches Claim 4.2 (ii), and hence Claim 4.2 and the lemma follows. $\square$

Lemma 4.6 (Truth of outcome lemma) *Let n be any ball. Then either n is a stable x -ball or n adheres to a stable x -ball.*

Proof. By Lemmas 4.4 and 4.5. $\square$

Lemma 4.7 $C_F(A) \cong B$.

Proof. The desired isomorphism is induced by the injection from $A \rightarrow B$ given by $\hat{x}_j \rightarrow x_j$. The lemmata and addition of y points achieve the rest. $\square$

As we see in Sections 5 and 7, Watnick's theorem has a number of important extensions and generalisations. Along related lines, Lerman [138] also used an infinite injury argument controlled by a tree to show:

Theorem 4.8 (Lerman [138], Theorem 2.2) *If A is an infinite Σ_3 set, then there is a computable linear ordering B of order type*

$$\zeta + n_0 + \zeta + n_1 + \cdots$$

with $n_0, n_1, \dots$ listing A in order of magnitude.

(Here, and henceforth, we assume $0 \notin A$.)

We shall call the representation of A in Theorem 4.8 a *strong ζ -representation*. If this representation is not in order and perhaps has repetitions, we shall call it merely a *ζ -representation*. Note that by Theorem 4.8 and counting quantifiers, A has a strong ζ -representation iff A has a ζ -representation iff A is Σ_3 .

We can similarly define (strong) η -representation. These have been investigated by Lerman [138], Feiner [67], Fellner [71] and Rosenstein [208].

Theorem 4.9

- (i) (Feiner [67]) *If A has an η -representation then A is Σ_3 (see Theorem 2.5).*
- (ii) (Rosenstein [208]) *If A has a strong η -representation then A is Δ_3 .*
- (iii) (Rosenstein [208]) *Every Σ_2 set has a strong η -representation.*
- (iv) (Fellner [71]) *Every Π_2 set has a strong η -representation.*

Proof.

(i) and (ii) come from looking at the quantifier form. For instance, for (ii), by (i) we know if A has a strong η -representation L then A is Σ_3 . So it suffices to show that it is also Π_3 . We need only show $\bar{A}$ is Σ_3 for this. Now $n \in \bar{A}$ iff there are $x_1, \dots, x_{n+1}$ with (x_i, x_{i+1}) a successivity if $i \neq j$ such that for all distinct $y_1, \dots, y_n$ with all the $y_i \leq x_1$ for all i there is an i, j and a z with $y_i < z < y_j$. Clearly this is Σ_3 and defines $\bar{A}$ as L is a *strong* representation.

(iii) Let A be Σ_2 . So for some computable R we have $x \in A$ iff $(\exists y)(\forall z)(R(x, y, z))$. By a standard trick we can choose R so that there is $X \in A$ iff there is a *unique* y such that for all z , $R(x, y, z)$. We then construct intervals $I_0 < I_1 < \dots$ so that the desired ordering is $\sum_x I_x$. We subdivide I_x into intervals $\{J_{\langle x, i \rangle} : i \in \omega\}$ so that $I_x = \sum_y J_{\langle x, y \rangle}$. Then $J_{\langle x, y \rangle}$ will either have order type x or η . We start with $J_{\langle x, y \rangle}$ of type x until we see some z with $\neg R(x, y, z)$. We then make $J_{\langle x, y \rangle}$ have type η . This evidently works.

The proof of (iv) can obviously be assembled along lines similar to Watnick's or Lerman's by taking a Π_2 A represented by a tree and then using this to place points. We sketch this below.

Let A be Π_2 so that $n \in A$ iff $(\forall x)(\exists y)(R(x, y, n))$ with R the relation representing the true path on a tree. Without loss of generality $|A| = \infty$. We sketch the argument for two numbers. Suppose 2 and 4 are the least two numbers in A . The construction begins by trying to build an ω sequence

$$x_0^s \ x_1^s \ x_2^s \ \dots$$

When we see n appear to be the least number in A we must chop the above down to n . That is, we will ensure that a dense piece follows a block of n things. As $2 = \mu z (z \in A)$ we need to construct this so that the final type is

$$2 + \eta + \dots$$

Now 1 may appear in A finitely often. Each time it does so, we can cancel any work we have done based on the belief that $2 = \mu z (z \in A)$. In particular, we may cancel the block x_0^s, x_1^s and put points between x_0^s and x_1^s building the beginning of an η -sequence. However at some stage t we will have that, for all $x > t$, $1 \notin A_s$. Then at t we will ensure that $(\forall t_1 > t)(x_0^t = x_0^{t_1}$ and $x_1^t = x_1^{t_1})$.

Now (after t_1) suppose we next see a stage t_2 where $2 \in A_{t_2}$ and $3 \in A_{t_2}$. Then we will have to attempt

$$x_0 \ x_1 \ \eta \cdot z_0 \ z_1 \ z_2$$

To do this we block off z_0, z_1, z_2 and agree that

- (a) this will be *part of* a block based on the guess $2 \in A$
and $(\forall t_3 \geq t_2)(1, 0 \notin A_{t_3})$, and
- (b) this will be a complete block if *additionally* $3 \in A$.

To make (a) and (b) compatible, we need only ensure that if $3 \notin A$ then $z_0 z_1 z_2$ is incorporated into some larger block.

Later, as stage u , we may see $2 \in A_u$, $3 \notin A_u$, $4 \in A_u$. Then we would densify between $x_0 x_1$ and the $z_0 z_1 z_2$ blocks regarding the point currently following z_2 as part of a 4-block. A typical situation is

$$x_0 x_1 p p p p p z_0 z_1 z_2 q q q q$$

becoming

$$x_0 x_1 r p r p r p r p r p r p r z_0 z_1 z_2 q r q r q r q r$$

where r represents the new points. Then if later we see $3 \in A_u$, we'd add a new point between z_2 and q .

In this way we will build the correct order type corresponding to the true path of T . $\square$

A classification of the η -representable A seems quite hard. The depth of the problem was revealed by the following result of Lerman.

Theorem 4.10 (Lerman [138]) *There is a Δ_3 A such that A is not η -representable.*

Proof. The technique here is rather different from previous arguments and is an 'oracle' construction. Let D be Δ_3 complete and $L_0, L_1, \dots$ list the computable linear orderings. We ensure that $A \leq_T D$ so that A is Δ_3 . The requirements we meet are

$$R_e : L_e \text{ does not } \eta\text{-represent } A.$$

Let $[a, b]_e = \{x \in L_e \mid a \leq x \leq b\}$. At stage s , of the construction we perform the following steps:

STEP 1: For each $i < s$ which is *active* via an interval $[a_i^s, b_i^s]$ at the end of $s - 1$, ask D if $[a_i^s, b_i^s]$ is a complete block of cardinality s . If the answer is no for all $i < s$, put s into A . If the answer is yes for some i let $i = i(s)$ and put s in $\bar{A}$. Declare i as inactive.

STEP 2: If $i(s)$ is defined in step 1, cancel the assignment of $[a_i^s, b_i^s]$ to L_i for all $i \geq i(s)$.

STEP 3: Fix the least i with L_i active, yet no interval assigned to L_i . Ask D if L_i is total. If L_i is not total declare L_i as inactive (and go to step 4).

Ask D if L_i has points a, b with $s+1$ points between them. If not, declare L_i as inactive and go to step 4. If the answer is yes, fix a finite interval $[a_i^s, b_i^s]$ that is not a complete block and has cardinality s and assign it to L_i .

STEP 4: For each $i \leq s$, if there is an interval currently assigned to L_i that is not maximal, use D to construct $[a_i^{s+1}, b_i^{s+1}]_i \supsetneq [a_i^s, b_i^s]$ of L_i of cardinality $s+1$. This is now assigned to L_i in place of $[a_i^s, b_i^s]$.

By construction $A \leq_T D$. Note that if $i(s)$ is defined then $i(s+1) < i(s)$ and hence $|A| = \infty$. To see that L_i does not η -represent A , for an induction let s_0 be a stage where the L_j for $j < i$ are not activated after stage s_0 . Then if $[a_i^s, b_i^s]$ is assigned to L_i after s_0 (choosing s_0 least) we see that as this assignment is not cancelled, if L_i is not inactivated then $\bigcup_{s>s_0} [a_i^s, b_i^s]$ is an infinite block in L_i , so L_i cannot represent A . On the other hand, if i is inactivated at $t > s_0$ then L_i has a complete block of size t , yet $t \notin A$. The result follows. $\square$

Lerman did note that although not all Σ_3 sets can be realized by η -representations, all such *degrees* can be.

Theorem 4.11 (Lerman [138]) *Let A be Σ_3 . Let*

$$B = A \oplus \omega = \{2x : x \in A\} \cup \{2x+1 : x \in \omega\}.$$

Then B is η -representable. Since $B \equiv_T A$ it follows that all Σ_3 degrees can be η -represented.

Proof. The argument is a simple coding one. If A is Σ_3 , we know for some computable R that $x \in A$ iff $(\exists y)(\forall z)(\exists t)(R(x, y, z, t))$. For each $e \in \omega$ we construct an interval I_e and then the required ordering will be $L = \sum_{e \in \omega} I_e$.

Let $e = \langle x, q \rangle$. We build $I_{\langle x, q \rangle}$ as follows. If x is odd we simply make $I_{\langle x, q \rangle}$ of type $\eta + x + \eta$. If x is even we begin by having the interval of type $\eta + x + 1 + \eta$. If we see $R(x, q, 0, s)$ hold for some s , we say that 0 is verified for (x, q) and absorb the 1 into an η piece so that for one stage, $I_{\langle x, q \rangle}$ has type $\eta + x + \eta$. At the next stage, we put a new 1 back in and have $\eta + x + 1 + \eta$ again until 1 is verified i.e., $R(x, q, 1, t)$ also holds for some t . It is easy to see that L has the desired properties. $\square$

Lerman notes that Theorem 4.11 can easily be modified to work for any Σ_3 set B that is not immune. (That is, B has an infinite computable subset.)

Question 4.1 *Is there a similar classification of strongly η -representable Δ_3 degrees? In particular, is each Δ_3 degree strongly η -representable?*

We close this section with a brief look at another special class of order types. Let $f : \mathbb{Q} \rightarrow \mathbb{N} - \{0\}$. We shall say an order type $\mathbf{A}$ is η -like if it has the form $\sum\{f(q) : q \in \mathbb{Q}\}$. We shall say $\mathbf{A}$ is *strongly η -like* if the range of f is finite. Because of their simplicity and because they turn up in many contexts (as we will see and have seen) η -like orderings have attracted quite a bit of attention. We restrict ourselves in this section to studying the complexity of f if $\mathbf{A}$ is a computable (strongly) η -like ordering, and conversely. The first result here is

Theorem 4.12 (Rosenstein [208]) *If $\mathbf{A}$ is computable and η -like then there is a Δ_3 function g such that $\mathbf{A} \cong_{\text{comp}} \sum\{g(q) : q \in \mathbb{Q}\}$.*

Proof. We need to build $g \leq_T \mathbf{0}''$. We do so as follows. Let $q_0, q_1, \dots$ list $\mathbb{Q}$, and $a_0, a_1, \dots$ list A . At stage s we define $g(q_s)$. Define $f(q_0)$ as follows: Ask the $\mathbf{0}''$ -oracle if there is an $a_i \neq a_0$ that is either the successor or predecessor of a_0 . Note that $\mathbf{0}''$ can answer such questions. In this way, we know that all blocks are finite, and we can compute $B(a_0)$, the block containing a_0 . Now define $g(q_0) = |B(a_0)|$. Now continue in the obvious way. If $q_1 < q_0$ work below $B(a_0)$ and if $q_0 < q_1$ work above $B(a_0)$. If, for example, $q_1 < q_0$, find the least (by Gödel number) $a_i \notin B(a_0)$ and similarly define $g(q_1) = |B(a_i)|$. Now the density of $\mathbb{Q}$ ensures that this can be extended to an isomorphism and g is Δ_3 by construction. $\square$

Using an approximation argument similar to, for example, Theorem 4.9 (iii) and essentially the tree representation of a Π_2 set, Fellner showed

Theorem 4.13 (Fellner [71]) *If f is Π_2 then $\sum\{f(q) : q \in \mathbb{Q}\}$ is computably presentable.*

Proof. Let $x \in A$ iff $(\forall z)(\exists y)(R(x, z, y))$, where $A = \{\langle q, n \rangle : f(q_1) = n\}$ when $R(\langle q_1, n \rangle, z, y)$ holds. (We can suppose that $\neg R(\langle q, m \rangle, z, y)$ for all $m > n$, and say that $\langle q, n \rangle$ looks correct at stage y .) Remember that if $x, y \in A$ then x and y appear *together* on A infinitely often. Now we define a linear ordering B and a partial function f_s from $\mathbb{Q}$ to the blocks of A so that $f = \lim_s f_s$. Define $f_0(q_0) = n_0$ with n_0 such that $R(\langle q_0, n_0 \rangle, 0, y)$

holds for some least y and put down a block of n_0 ones, that are labelled $\langle q_0, n_0 \rangle$. This label remains unless for some $s > y$ and some $n_i < n_0$ we see $R(\langle q_0, n_i \rangle, 0, s)$ hold in which case we release the last $n_0 - n_i$ ones (to be put into other blocks). If at some stage $t > y$ we instead see $R(\langle q_0, n_j \rangle, 0, t)$ hold for some $n_j > n_0$ then we begin a block of n_j ones *containing* the n_0 ones (and have $f_t(q_0) = n_j$). It is easy to see that after a finite number of failures, if $\langle q_0, n \rangle$ is the *true* member of A , the n ones put down will be a block that will never be split. Of course, for infinitely many s we may build blocks with $n_{i_s} > n$ many ones, but whenever n looks correct, they will be split off from the n block.

Strategies are combined in the obvious ways as in, for example, Watnick's theorem, so if we have for $\langle q_1, k \rangle$ guesses as to the correct $\langle q_0, n \rangle$, when these prove wrong, elements can be released to the construction. The only care that is really needed is to ensure that any one that is put down eventually adheres to some block, and this is achieved by using the least element that is around as building materials for blocks. $\square$

Similar arguments show

Theorem 4.14 (Downey) *If f is Σ_2 then $\{\Sigma f(q) : q \in \mathbb{Q}\}$ is computably presentable.*

Sketch Proof. Now $f(q) = \dot{n}$ iff $(\exists x)(\forall y)(R(\langle q, n \rangle, x, y))$. We map q_0 initially to the first n_0 such that we see $R(\langle q_0, n_0 \rangle, x, 0)$ hold for some x . We keep q_0 mapped to n_0 until we see (if ever) $R(\langle q_0, n_0 \rangle, x, s)$ fail to hold at some stage s . We then switch to the least n_i such that for some least x , for all $j \leq s$, $R(\langle q_0, n_i \rangle, x, j)$ holds. If $n_i < n_0$ we release those ones in $n_0 - n_i$. If $n_i > n_0$ we build the block around the n_0 block. As there really is some n with $f(q_0) = n$ we eventually settle on $f(q_0) = n$ for some n . The inductive strategies combine in a straightforward way. $\square$

In view of Theorem 4.12, a natural question is whether the above results can be improved to Δ_3 . Lerman and Rosenstein [140] showed that the answer is no.

Theorem 4.15 (Lerman and Rosenstein [140]) *There is a Δ_3 function f such that $\sum\{f(q) : q \in \mathbb{Q}\}$ is not computably presentable.*

Proof. The method is an oracle construction like Theorem 4.10. So let $\{L_e : e \in \omega\}$ list all computable linear orderings. We build f computably in $0''$.

To show that L_e does not represent f , we proceed as follows. First $0''$ can decide if L_e is total. If this is not true then we need do nothing. So we henceforth assume L_e is total. Now we will ensure that for each n there is at most one $q \in \mathbb{Q}$ with $f(q) = n$. As we go along in the construction we will build blocks. Suppose we have an n block. We ask $0''$ if L_e has two n -blocks for $m \geq n$. If not we define $f(q) = n$, and $f(r) = n$, some q, r and need do nothing more. If L_e has two m -blocks for $m \geq n$, we will find two such sets of n successivities $B_{e,s}$ and $\hat{B}_{e,s}$. We then use these as follows. At each future stage we ask if $B_{e,t}$ and $\hat{B}_{e,t}$ are complete blocks. When the answer is no, for each that is not complete, find, for example, $B_{e,t+1} \supsetneq B_{e,t}$.

In this way we can eventually get an infinite block in L_e (and so we really do nothing for L_e) or eventually we can find complete blocks B_e and $\hat{B}_e$ with $|B_e| \neq |\hat{B}_e|$ and we have ensured that $|B_e|, |\hat{B}_e|$ are not yet in $\text{ran } f$. Then if $B_e < \hat{B}_e$ in L_e we define $f(q) = \hat{B}_e$ and $f(r) = B_e$ for some $q < r$ in $\mathbb{Q}$. This makes the order type of L_e wrong. $\square$

We remark that Theorem 4.15 can be improved to construct f with bounded range so that $\sum\{f(q) : q \in \mathbb{Q}\}$ would be strongly η -like.

Theorem 4.16 (Downey) *There exists a Δ_3 f with range $\{1, 2, 3, 4\}$ such that $\sum\{f(q) : q \in \mathbb{Q}\}$ is not computably presentable.*

Sketch Proof. This time the strategy is different. First we define $f(q) = 4$ for all $q \in \omega$ with $q > 1$. We then define $f(q) = 2$ for all $q \leq 0$ with $q \in \mathbb{Q}$. The intervals between these blocks, we will either make of type $\eta + 3 + \eta$ on η , depending on how we wish to diagonalise against the L_e 's. We use the interval between $f(e)$ and $f(e+1)$ to kill L_e . For L_0 we ask $0''$: does L_0 have a block of 4 points? If yes, find one and ask $0''$ if it is complete. If yes, call this $B_{0,0}$. If no, forget L_0 and make the interval $(f(0), f(1))$ of type η . Now ask $0''$: does L_0 contain a 4-block to the left of $B_{0,0}$.

- (i) If yes, define $f(r) = 1$ for the least (by Gödel number) $r \in (0, 1)$. Find such a block $B_{0,1}$ to the left of $B_{0,0}$ and now repeat the process above, with $B_{0,1}$ in place of $B_{0,0}$.

- (ii) If no, ask $0''$ if L_0 contains a 3-block to the left of $B_{0,0}$ or, more generally, $B_{0,n}$. If yes, it must be a complete 3-block and we can win by making $f(q) = 1$ for all $q \in (0, 1)$. If no, then we can pick some $q \in (0, 1)$, with $f(q)$ as yet undefined, and define $f(q) = 3$ with $f(r) = 1$ otherwise, for $r \in (0, 1)$. This kills L_0 , since if the second option never pertains at any step n , we ensure that the interval $(0, 1)$ has type η and L_0 is wrong as it has an ω^* sequence of 4-blocks.

For L_1 we work similarly except now we try to find the 2 least 4-blocks and work between them. The extension to L_e is obvious and we leave the remaining details to the reader. $\square$

The key gap left open from the Fellner-Lerman-Rosenstein results is whether Theorem 4.13 has a converse; that is, can Theorem 4.12 be improved from Δ_3 to Π_2 ? This question was noted in several places such as Rosenstein [208], Fellner [71], Lerman-Rosenstein [140].

5 Effective Content of Some Classical Theorems: Subsequences, Embeddings and Automorphisms

In this section, we shall pursue one of the themes we mentioned earlier: investigating the effective content of classical theorems. Here the classical theorem (or fact) might say: every $(*)$ ordering of type A has a $(*)$ subordering of type B. The idea is to replace the $(*)$ by the word ‘computable’ and see if the result remains true. If the result is now false we might then see what we can salvage.

We begin by looking at suborderings. The earliest result here is due to Tennenbaum (see Rosenstein [208]) and independently Denisov (see Goncharov and Nurtazin [86]). It is easy to see that each infinite linear ordering has an infinite subordering of order type either ω or ω^* . In contrast, we have

Theorem 5.1 (Tennenbaum, Denisov) *There is a computable linear ordering of order type $\omega + \omega^*$ with no infinite computably enumerable suborderings of order type ω or ω^* .*

Proof. The proof is by a finite injury priority argument. Since we have already looked at this technique in some detail, it will suffice to describe the

strategies for single requirements. We build $A = \cup_s A_s$ in stages to satisfy the requirements.

N_e : a_e either has finitely many predecessors, or finitely many successors.

R_e : If W_e is infinite, then it is not an ω sequence.

R_e^* : If W_e is infinite, it is not an ω^* sequence.

To meet the requirement N_e , at some stage $s = s(e)$ we need to declare some finite sequence $a_{i_1}, \dots, a_{i_k}$ such that either $a_{i_1}, \dots, a_{i_k}, a_e$ is an initial segment of A , or $a_e, a_{i_1}, \dots, a_{i_k}$ is a final segment of A .

The basic module for R_e is this. We wait till we see some *unrestrained* a_i occur in $W_{e,s}$ (for $i > e$). (Note that this is a finite injury argument and we ensure that the N_j , R_j and R_j^* for $j < e$ restrain only finitely many elements: if a_i fails to occur then $|W_e| < \infty$ so we win.) When $a_{i,s}$ occurs, we declare a_i and all of its current successors to be a final segment of A with priority e . For instance, we might have

$$a_0 a_1 \cdots a_j a_i a_{i_1} \cdots a_{i_k} = A_s.$$

Then $a_i a_{i_1} \cdots a_{i_k}$ is declared as restrained with priority e so that, with priority e we add no new elements to this set. Note that if we succeed with this restraint then W_e cannot be an infinite ω sequence as it contains a_i . The net effect is that a_{s+1} would be placed on the board between a_j and a_i as would subsequent a_t until some other R_k (R_k^*) acts. The R_e^* work dually and the N_e simply assert 'whatever a_e 's current affiliation is (i.e., final or initial segment) preserve this'. The details fit together using a standard application of the finite injury argument. $\square$

It is not difficult to modify the above to show that A can be constructed so that both initial segments and final segments of A are hyperimmune. Recall that B is called hyperimmune if there is no infinite disjoint collection of canonical finite sets $\{D_{g(x)} : x \in \omega\}$ such that for all x , $D_{g(x)} \cap B \neq \emptyset$. Recall that a set D is called *cohesive* if there is no r.e. set Q with $|D \cap Q| = |D \cap \bar{Q}| = \infty$. It is also not difficult to show that we cannot improve Theorem 5.1 to make the ω or ω^* pieces cohesive (see Soare [232, ch. X, Exercise 3.12]. In fact we cannot make the ω or ω^* pieces hyperimmune (Martin, see Jockusch [102, Corollary 4.5]). Here B is called hyperimmune if there is no disjoint r.e. collection of r.e. sets (a *weak array*) $\{W_{f(x)} : x \in \omega\}$ such that for all x , $W_{f(x)} \cap B \neq \emptyset$.

The Tennenbaum-Denisov theorem has been generalised in many ways. One of the earliest was the original motivation for Watnick's theorem, which we met in Section 3. Recall that there we said that Watnick showed that τ is a Π_2 order type iff $\zeta\tau$ is computably presentable. Watnick proved the following extension.

Theorem 5.2 (Watnick [243]) *If τ is any Π_2 order type, there is a computable linear ordering A of type $\omega + \zeta\omega + \omega^*$ with no computable ω or ω^* sequences.*

Proof. The idea is to take the proof of Section 3 and add a further layer of requirements

R_e : W_e is not an ω sequence.

R_e^* : W_e is not an ω^* sequence.

The details are a little complex, so we will only sketch how to combine the R_e 's above with the Section 3 construction as meeting the requirements

M_e : if $a_e \in \tau$ put an ω^* and ω sequence around a_e .

These in turn can be split into

$M_{e,i}$: if $a_e \in \tau$ put at least i immediate predecessors
and i immediate successors around a_e .

The idea is as follows.

To meet, for example, R_e we try to force W_e to contain a member of the final ω^* sequence. During the construction around (apparent) points of τ we will be building partial blocks of points. For the sake of R_e , if we see some x occur in a block of a_j for $j > e$ ² then we can abandon totally this version of a_j (by priorities) and if, for instance, there is no a_i for $i < e$ between the a_j and the final ω^* region, simply add the a_j block (at s) to the ω^* block and pick a new a_j point around which to build a ζ block. See Figure 1 below for a typical scenario.

²Naturally, in the Π_2 version we would look for x in the block a_μ with μ of low priority than σ with σ the guess of R_e .

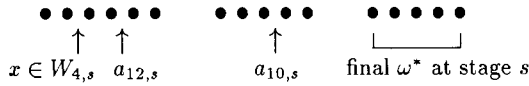


Figure 1.

Here we see $x \in W_{4,s}$ and only the apparent positions of $a_{12,s}$ and $a_{10,s}$ between x and the ω^* portion. We'd then get, at stage $s + 1$, the situation of Figure 2.

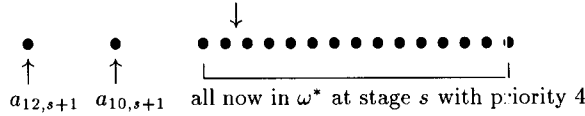


Figure 2.

The problems all stem from the fact that we could not use the idea above if we had $a_{3,s}$ instead of $a_{12,s}$ or $a_{10,s}$ as these have higher priority than R_4 , so R_4 ought not to be able to erase them. The idea is then as follows. Suppose, for instance, we had $a_{3,s}$ in place of $a_{10,s}$. When we see x in $W_{4,s}$ we'd do the best we can: we'd add the $a_{12,s}$ block (including x) to the $a_{3,s}$ block to get the situation of Figure 3 at stage $s + 1$.

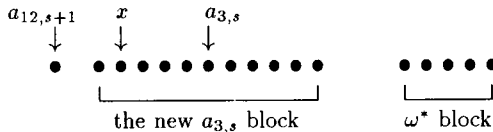


Figure 3.

Note that since x is to the left of $a_{3,s}$ and part of its block, if W_e contains an ω sequence, infinitely many $x' > a_{3,s}$ must occur in $a_{3,s}$'s block. *This is the key to the whole construction*; we wait till some x' occurs in $W_{e,s}$ and $a_{3,s}$'s block more than (say) $3 + 4 = 7$ further on than $a_{3,s}$ as in Figure 4.

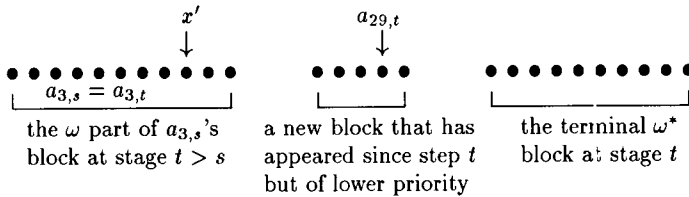


Figure 4.

We would then add all the points from x' onwards to the final ω^* block at stage $t + 1$ with priority 4.

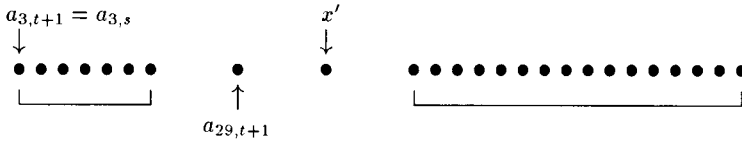


Figure 5.

The whole point is that such actions can only affect the ρ block around $a_{3,s}$ further and further out (i.e., of lower and lower priority) and so only finitely often injure some $M_{e,i}$.

The remaining details are rather messy, but one can fit all of the above together on a tree and get the desired results. $\square$

Another generalisation of Theorem 5.1 goes back to the thesis of Hird [96] which appeared in [97]. One can view the Tennenbaum-Denisov theorem as constructing something bad isomorphic to $\omega + \omega^*$. We note that $\omega + \omega^*$ is a convex linear ordering.

Theorem 5.3 (Hird [96]) *Let A be a computable linear ordering with end points with P a co-r.e. subset. Then the following are equivalent.*

- (i) *There is a computable linear ordering $B \cong A$ with the image of P not r.e..*
- (ii) *The subset P does not have both infimum and supremum in A .*

Proof. Actually we prove a slight extension of Hird's result, since he assumed that P was co-r.e. in A . First if $(a, b) = P$ then P is clearly r.e.. Furthermore, as P is co-r.e., if P is r.e. then P is computable. So to complete the proof, suppose that A has a convex computable subset P with no supremum (say). We need to build $f : B \cong A$ with $f^{-1}(P)$ not r.e..

The basic idea is fairly familiar. We have a partial isomorphism

$$f_s : \{b_0, \dots, b_{n(s)}\} \rightarrow \{a_0, \dots, a_{n(s)}\}.$$

We wait till we see some $b_i \in W_{e,s}$ with $f(b_i) \in P_s$. We then wish to remove b_i from $f^{-1}(P_s)$. This is okay since P_s has no supremum. We need only redefine

the isomorphism so that it takes b_i (and all of its successors at stage s) to elements out of P (which we can do as P is computable and as A has end points) so that there are infinitely many elements $> P$ in A .

To make the argument valid we add priorities. That is, we don't allow our action for W_e to affect the $f_s(b_0), \dots, f_s(b_e)$. $\square$

One can apply the theorem above to the set

$$P = \{x : x \text{ has finitely many predecessors}\}$$

in $\omega + \omega^*$ to make it not co-r.e.. Indeed it is again easy to make it *immune* of any r.e. degree of unsolvability (this gives the Tennenbaum-Denisov theorem) or similarly to show

Theorem 5.4 (Hird [96]) *Let P be a co-r.e. convex subset of A . There is a computable $B \cong A$ via f with $f^{-1}(P)$ immune iff P satisfies one of the following:*

- (i) P has order type ω and no supremum in A .
- (ii) P has order type ω^* and no infimum in A .
- (iii) P has order type $\omega^* + \omega$ and neither infimum nor supremum in A .

We remark that Hird obtained similar results for P being hyperimmune and extended Martin's result to show that

Theorem 5.5 (Hird [96]) *No convex subset of a computable linear ordering can be hyperimmune.*

Proof. If P does not have order type ω , ω^* or $\omega + \omega^*$ and P is convex, then P cannot be (hyper)immune. Then we need only show the theorem for P of order type ω , ω^* or $\omega^* + \omega$. So, for instance, choose P of type ω . We build a weak array $\{C_e : e \in \omega\}$ so that for all e we meet

$$R_e : C_e \cap P \neq \emptyset.$$

Let p be the least element of P and $b_0, b_1, \dots$ list $\{x : x \in A \text{ and } x \geq P\}$. At stage s choose the least e such that there is no element to the left of b_{s+1} in $C_{e,s}$ and add b_{s+1} to $C_{e,s}$. To see that all the R_e are met, assume for an

induction that this is so for all $i < e$ and additionally assume that each C_i is finite. Let s_0 be a stage where $C_{i,s_0} = C_i$ for all $i < e$. Let $s_1 > s_0$ be the first stage where $b_{s_1} \in P$. Then either b_{s_1} is added to C_{e,s_1} or there is already some $b_j < b_{s_1}$ in C_{e,s_1} . It is clear that as there are only finitely many b_k with $P \leq b_k \leq b_j(b_{s_1})$ that $|C_e| \leq |C_{e,s_1}| + b_{s_1}$. $\square$

We remark that these results were only a small part of Hird's thesis which was devoted to analysing intrinsic properties of the lattice of relations of a computable structure under weak decidability conditions. Many known algebraic theorems can be deduced from Hird's model theoretic work.

Going back to the original observation about ω and ω^* sequences, we can salvage the following effective version.

Theorem 5.6 (Rosenstein [208]) *If A is a computable linear ordering, then A has a recursive subordering of type ω , ω^* , $\omega + \omega^*$, or $\omega + \zeta\eta + \omega^*$.*

Proof. If A has no ω nor ω^* computable sequence it must have a first and a last element. With the exception of these, each element must have a successor and a predecessor. Thus the order type of A must be $\omega + \zeta\alpha + \omega^*$ for some α . If $\alpha \neq \eta$, it is easy to see that A has a computable subordering of type $\omega = \omega^*$, giving the theorem. $\square$

Rosenstein [208] asked if $\omega + \zeta\alpha + \omega^*$ was necessary in Theorem 5.6. Lerman answered this question affirmatively.

Theorem 5.7 (Lerman [138]) *There is a computable linear ordering with no recursive subordering of type ω , ω^* or $\omega + \omega^*$.*

Proof. The argument is a straightforward Π_2 argument rather along the lines of a maximal set construction. We will confine ourselves to the technique used to meet two requirements, and leave the inductive strategies to the reader. We meet the requirement

R_e : Either W_e is infinite, or W_e contains an element with infinitely many predecessors and successors (in W_e).

To meet R_0 we specify that a_0 is the least element of A , a_1 the greatest, and work in (a_0, a_1) . R_0 does nothing until some $x \in W_{e,s}$ with $x \neq a_0, a_1$, so that $x \in (a_0, a_1)$. R_0 now freezes the interval (x, a_1) and specifies that the

ordering is built in (a_0, x) until some y occurs in W_e with $y_0 \in (a_0, x)$. R_e will then switch to build the ordering in (x, a_1) until some z_0 occurs in (x, a_1) and then R_e switches back to (a_0, x) etc.. Then if R_e acts infinitely often x will have infinitely many elements in W_e on each side. So when R_0 says 'work in (a_0, x) ', R_1 can only pick some $z \in W_{1,s}$ which occurs in (a_0, x) . If R_1 switches to (x, a_1) we leave this version of R_1 on hold and start R_1 again in (x, a_1) . If R_0 switches back, we can return to the old version of R_1 . So if W_0 is infinite, R_0 imposes essentially no restraint on R_1 . If W_0 is finite then R_0 imposes only finitely much restraint on R_1 .

The inductive strategies for $n > 2$ requirements contain no surprises and the details follow as usual. $\square$

It is also possible to give an effective version different from Theorem 5.6.

Theorem 5.8 (Manaster) *If A is an infinite computable linear ordering, then A has a Π_1 subset of type ω or ω^* .*

Proof. Assuming A has no infinite computable ω or ω^* sequence, it has a least element a_0 . We will define collections $y(i, s)$ and $z(i, j)$. We let $y(0, s) = a_0$ for all s . At stage 1 we let $y(1, 1) = a_1 = z(1, 1)$. We keep $y(1, s) = a_1$ until a stage s_1 occurs (it may not) such that there exists x with $a_0 < x < a_1 = y(1, s_1)$. We then define $y(1, s_1 + 1) = x$ and $z(1, 2)$. Now continue in the obvious way. Thus either for some s_n , for all $s > s_n$ we have $y(1, s) = y(1, s_n)$, or for all s there is a $t > s$ with $y(1, t) < y(1, s)$. In the latter case, $\{z(1, i) : i \in \omega\}$ is a computable ω^* sequence.

The inductive strategies are similar. We want to define $y(2) = \lim_s y(2, s)$ (if it exists). At each stage s we assume that the correct position of $y(1, s)$ is final. We must follow only two rules; we must always work right of $y(1, s)$ and never let $y(2, s) = y(1, t)$ for any $t \in \omega$. Note that in general, either there is an i with $\lim_s y(i, s)$ failing to exist (so an ω^* sequence) or, for all i there is an s_i with $y(i, s) = y(i, s_i)$ for all $s > s_i$ and so a Π_1 ω sequence consisting of $\{y(i) : i \in \omega\}$. $\square$

Actually the *proof* above shows that

Corollary 5.9 (Downey) *If A is a computable linear ordering, then if A does not have a Π_1 ω (resp. ω^*) sequence, it has a computable ω^* (resp. ω) sequence. In particular, if A contains no computable ω nor ω^* sequences then it has both a Π_1 ω and a Π_1 ω^* sequence.*

One can well ask what can be said of the *individual* suborderings in the ordering L . For instance, suppose that the ordering has an infinite ω subordering. How hard can it be to find such an ordering? The answer to this question is, perhaps, slightly surprising. We need a definition. The arithmetical hierarchy can be extended by allowing set quantifiers.

Let R be a relation and $S \subseteq \mathbb{N}$. Then we can say that $R(S)$ holds iff for all initial segments $\bar{x}$ of S we have that $R(\bar{x})$ holds. Then we define a set A to be Σ_1^1 iff there is a arithmetical R such that

$$x \in A \text{ iff } (\forall S) R(x, S).$$

We say that A is Π_1^1 iff $\bar{A}$ is Σ_1^1 , and extend to Π_n^1 , Δ_n^1 and Σ_n^1 in the obvious way. We say that a set is *hyperarithmetical* if it is Δ_1^1 . (Kleene was the first person to study such sets.) Note that Π_1^1 sets are immensely complicated and can, in particular, compute all the n -th jumps of $\mathbf{0}$. Of interest to us is the following.

Theorem 5.10 (Kleene, see, e.g., Sacks [214]) *There is a computable linear ordering with an infinite descending sequence, but no hyperarithmetical such sequence.*

Sketch Proof. First one needs the classical result that there is a computable tree T of strings of integers with an infinite branch yet no hyperarithmetical branch (see e.g., Rogers [207]). T can be linearised by the Kleene-Brouwer ordering $\leq_K$ as follows. For strings σ we have $\sigma \hat{u} <_K \sigma \hat{v}$ if $v < u$ (note the inversion). We extend this to the whole tree by saying that $\sigma <_K \tau$ if either $\sigma \subseteq \tau$ or, failing that, there exist $\sigma' \subseteq \sigma$, $\tau' \subseteq \tau$ with σ' and τ' both of the same length and $\sigma' <_K \tau'$.

Note that each infinite branch of T defines an infinite descending sequence in the ordering L . Conversely, suppose that $x_0 > x_1 > \dots$ is a hyperarithmetical infinite descending sequence. Since sets of numbers of bounded length are clearly well ordered by $<_K$, we may as well suppose that the sequence $x_0, x_1, \dots$ has uniformly increasing length. It is not difficult to prove by induction on length that for all n , there is a j such that for all $i \geq j$, $x_j \subseteq x_i$. One can then inductively define a hyperarithmetical branch. $\square$

Actually, the above also leads us to a nice result about initial segments of linear orderings. Suppose that $A + B$ is a computable linear ordering. What does that say about the complexity of the initial segment A ? For instance,

must A have a computable copy? The answer is no in a very strong way. (We here assume that the reader is familiar with the recursive ordinals.)

Theorem 5.11 *Let R be a computable ordering with infinite descending sequence, yet no hyperarithmetical such sequence.*

- (i) (Gandy) *Let Q be the maximum well ordered initial segment of R . The Q is $\Pi_1^1 - \Sigma_1^1$, and has order type ω_1^{CK} , the first non-recursive ordinal.*
- (ii) (Harrison) *R has ordertype $\omega_1^{\text{CK}}(1+\eta)+\gamma$ where γ is a recursive ordinal.*

We omit the proof since it would take us a little far, but refer the reader to, e.g., Sacks [214, page 56], for the details.

Notice that the above leads to the slightly paradoxical observation that there are computable linear orderings $A+B$ such that the ordering $A+1+B$ has *no* computable presentation! (For to figure out if $x \in A$, one needs only compare with the point b representing the middle “1”.)

The above leads one to wonder if there are any conditions on A which guarantee computable presentability. In his thesis [187] Mathew Raw studied this question and used a variation of the Feiner technique to construct a computable $A+B$ such that A is Π_3 and has no computable presentation. On the other hand he proved that if $A+B$ is computable and has computable adjacency relation, then if A is Π_1 then it has a computable copy. This result was extended as follows.

Theorem 5.12 (Ambos-Spies, Cooper, Lempp [2]) *If $A \dashv B$ is computable, and A is Σ_2 then A is computably presentable.*

Sketch Proof. The proof is not difficult, but is a little delicate. First one proves that if I is a Σ_2 initial segment of L then there is a increasing Δ_2^0 sequence of elements of I cofinal in I . The proof is to use this sequence as boundaries for the Δ_2^0 isomorphism that is built. We build a computable $\langle M, < \rangle$. We use intervals (x_{n-1}, x_n) in I and map to (y_{n-1}, y_n) in M . This will be an isomorphism unless the intervals change. We will restart the map if possible unless we have already put more elements into the (y_{n-1}, y_n) than (x_{n-1}, x_n) , before they have settled down. Of course this is only a problem if the intervals are finite. The idea is to first “adjust” the isomorphism slightly to the left of y_n . If all the intervals are finite then the initial segment has type ω , which is fine. Otherwise, one is infinite and we can get to “catch up” the definition of the isomorphism. $\square$

Conversely, the following completely settles the exact quantifier level where non-presentability occurs.

Theorem 5.13 (Coles, Downey and Khoussainov [34]) *There is a Π_2 A with $A + B$ computable and such that A has no computable copy.*

Sketch Proof. The proof is a diagonalisation but is rather more subtle than ones we have seen so far. The technique below promises more applications. Suppose that we wished to enumerate the blocks of a ordering, and in this case we suppose that the ordering is η -like in the sense that it has no infinite blocks. Suppose that we had $\emptyset'$ as an oracle. Then we could determine successivities, and hence could demonstrate that if L is η -like, then $B(L)$ can be computed as the range of a $\emptyset'$ limitwise monotonic function. Here, a function f is *limitwise monotonic* if there exists a computable function $\varphi(x, s)$ such that

- (i) $\varphi(x, s) \leq \varphi(x, s + 1)$ for all $x, s \in \omega$,
- (ii) $\lim_s \varphi(x, s)$ exists for all $x \in \omega$,
- (iii) $f(x) = \lim_s \varphi(x, s)$.

The concept of a limitwise monotonic function is due to Khoussainov, Nies and Shore [116], and we have used the relativised notion. The crucial lemma is the following applied in relativised form.

Lemma 5.14 (Khoussainov, Nies and Shore) *There exists an infinite Δ_2^0 set S which is not the range of any limitwise monotonic function.*

Proof. The proof of the Lemma is a very easy diagonalisation argument which is an oracle construction. We must satisfy

$$R_e: S \not\subseteq \text{ra} \lim_s \varphi_e(\cdot, s) \text{ or } \varphi_e \text{ is not monotonic.}$$

The strategy for R_e is the following. We pick a witness x and put x into S . At each stage s we ask $\emptyset'$ if there is some $n \leq s$ such that $\varphi_e(n, s) = x$. If no such n exists then go to the next stage. Otherwise for each such n ask if there exists a stage $t > s$ such that $\varphi_e(n, t) \downarrow \neq x$. If the answer is yes, then x cannot be the pre-image of n by monotonicity. So we simply move

to the next unconsidered n , and if all such n are considered, simply go to the next stage. If the answer is no, then remove x from A , and cease all activity on R_e . Either $\varphi_e(x, t) \uparrow$ for some t , or $\varphi_e(x, t) = n$ for almost all t and $n \notin A$. $\square$

The proof of the theorem is complete via another coding argument.

Lemma 5.15 (Coles, Downey, and Khoussainov [34]) *If S is any Σ_3 set such that $m \in S \rightarrow m > 1$, then there is a computable $A + \omega^*$ such that A is η -like and $B(A) = S$.*

We will not prove Lemma 5.15 in any detail as it is a relatively difficult infinite injury priority argument rather along the lines of a combination of Feiner's original argument and a Watnick type argument, and we have already met arguments along these lines.

Basically, let S be given via

$$x \in S \text{ iff } (\exists n)(\forall y)(\exists z)R(x, n, y, z).$$

For each x as in Feiner's original theorem, we have to try to build a x block for each attempt at n . If we actually find a n such that the Π_2 condition $\forall y \exists z R(x, n, y, z)$ is satisfied then we build the x block. Otherwise we will kill the x block by either densifying it or have it incorporated in the final ω^* piece. For $x = 2$ and $n = 0$, we would begin with a 2-block $a_1 a_2$. While it looks bad, we regard it as part of the (finite segment built so far of the) ω^* block at the right end. We would be performing the rest of the construction to the left of this block. If at some stage, the Π_2 condition for $(2, 0)$ becomes verified another time, we would densify the work done (i.e., to the left of the 2-block) in the intervening stages (wrong guess), and begin work to the right of the 2-block, building a partial densification between this and the next block. Note that if this condition is infinitely often verified, then the material to the left will be dense, and other blocks will be built right of it. On the other hand if it is only finitely often verified, so either 0 is wrong or $x \notin S$, then the x will be incorporated in the ω^* piece. The inductive strategies work similarly except that we cannot densify things of higher priority than us. (The priority is given by putting all the pairs on a strategy tree.) The remaining details look like a Watnick style argument, and we refer the reader to the original paper for them.

To complete the proof, we can apply Lemma 5.15 to an S which is the range of no $\emptyset'$ limitwise monotonic function. Then A cannot be computably presentable as it is η -like. $\square$

A nice generalisation of the observation that every linear ordering has a subsequence from $\{\omega, \omega^*\}$ is due to Dushnik and Miller [61] who showed that an infinite countable linear ordering has a nontrivial self-embedding. (That is an isotone $f : A \rightarrow A$ such that f is not the identity.) The proof of this result runs as follows. If A has an interval of order type ω (or ω^*) then define f to be the identity outside of this interval and map i to $i + 1$ in the ω interval. If A fails to have such an interval then A embeds $\mathbb{Q}$. Now we can use Cantor's proof that all countable linear orderings embed into $\mathbb{Q}$ to do the rest.

The Dushnik-Miller result is not effective:

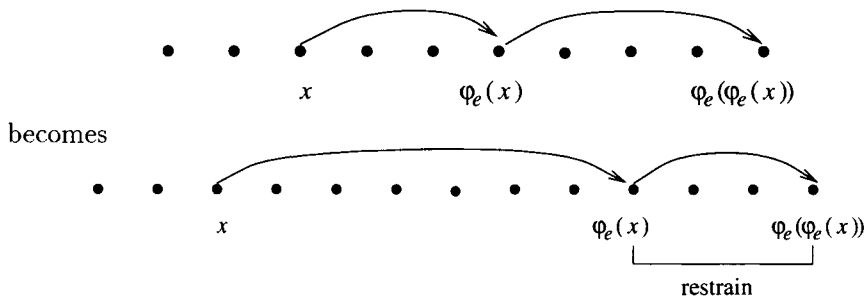
Theorem 5.16 (Hay and Rosenstein (in Rosenstein [208])) *There is a computable linear ordering of type ω with no nontrivial computable self-embedding.*

Sketch Proof. The proof is a finite injury priority argument. We build $A = \cup_s A_s$ so that we meet

N_e : a_e has $< \infty$ many predecessors

R_e : φ_e is not a nontrivial computable self-embedding.

To meet a single R_e , we wait until we see for some (unrestrained) x that $x \neq \varphi_{e,s}(x) \downarrow$ and $\varphi_{e,s}(\varphi_{e,s}(x)) \downarrow$. Note that if φ_e is to be a self-embedding, as we meet all the N_e , we need only worry if $x < \varphi_e(x) < \varphi_e(\varphi_e(x)) (< \dots < \varphi_e^n(x)$ for all n). The idea is simple. We simply add $|\varphi_e(x), \varphi_e(\varphi_e(x))| + 1 = n$ many new points between x and $\varphi_e(x)$ and then restrain $[\varphi_e(x), \varphi_e(\varphi_e(x))]$ with priority e . A typical situation is described below.



Then as $||[x, \varphi_e(x)]|| > ||[\varphi_e(x), \varphi_e(\varphi_e(x))||$, φ_e cannot be isotone and 1-1. It is now routine to continue the obvious strategies in the finite injury manner to a set (Theorem 5.16). $\square$

Actually, using a coding argument Downey and Lempp [57] have improved the above to read.

Theorem 5.17 (Downey and Lempp [57]) *There is a computable linear ordering L such that if f is a nontrivial self embedding of L then f can compute $\emptyset'$.*

A consequence of the Downey-Lempp Theorem is that the proof theoretical strength of the Dushnik-Miller theorem is ACA_0 .

Of course some computable order types *always* have non-trivial self-embeddings. For instance

Observation 5.1 (Folklore) *If A is a dense computable linear ordering, then A has a nontrivial computable automorphism.*

Proof. Use Cantor's back and forth argument. $\square$

Observation 5.1 points out the most natural self-embedding: an automorphism. Extending earlier work of Rosenstein, Lerman and others, Schwartz was finally able to classify linear orderings with nontrivial computable automorphisms. Define a linear ordering to be *computably rigid* if it has no nontrivial computable automorphisms.

Theorem 5.18 (Schwartz [219, 220]) *If A is computably rigid, then A has no interval of order type η . Furthermore, if τ is an order type containing computable linear orderings, and τ has no interval of type η , then τ contains a computably rigid linear ordering.*

Proof. Again we use a finite injury argument. Let A be an infinite computable linear ordering with no interval of type η . We construct $B \cong A$ via $f = \lim_s f_s$ so that we meet (for all $e \in \omega$)

R_e : φ_e is not a nontrivial automorphism of B .

To meet R_e we ensure that either φ_e is not total, or $\text{ra } \varphi_e \neq \omega (= A)$ or φ_e is not isotone and 1-1. Of course, as we go along, if we ever see $\dots b_i < b_j$ with

$\varphi_e(b_i) \not\leq_A \varphi_e(b_j)$ then we can stop as φ_e cannot be isotone. So we assume in the discussion below that this doesn't happen. The basic idea is simple and similar to Remmel's categoricity theorem.

Again suppose we knew the successivities of A . Then we'd wait till we saw some b_1, b_2 in B_s with $f_s(b_1) = a_1$, $f_s(b_2) = a_2$, and $\varphi_e(b_1) \neq b_1$ and (a_1, a_2) successivity. So we can wait until $\varphi_e(b_2) \neq b_2$. Now, without loss of generality, we can suppose $\varphi_e(b_1) > b_1$ so that $\varphi_e(b_2) > b_2$. The basic idea is then to split $(\varphi_e(b_1), \varphi_e(b_2))$ by adding a new element b_{s+1} to the interval spoiling φ_e as an automorphism. To do this, as usual, we need to revise $f_{s+1}(\varphi_e(b_2))$, and so need to be careful — as in Remmel's theorem — that we don't move $f_s(b_0), \dots, f_s(b_e)$. Also, as in there, we aren't able to identify the successivities in A , but know that each interval $[a, b]$ contains one. We simply guess then, as we did in Remmel's theorem, choosing the least one each time. The remaining details fit together in a familiar way via the finite injury method. $\square$

Lerman and Rosenstein [140] raised the question of the classification of order types τ with a nontrivial Π_1 automorphism for each computable copy in τ . Kierstead [121] used an infinite injury argument to construct a computable linear ordering of type 2η with no nontrivial Π_1 automorphism. This was extended to any order type $\sum\{f(q) : q \in \mathbb{Q}\}$ with $f(q) \in \mathbb{N} - \{0\}$, with no interval of type η , and f a Π_2 function of Downey and Moses [58]. In view of Theorem 4.16 we thus have:

Theorem 5.19 (Downey) *If A is an η -like computable linear ordering with no interval of type η , then there is a computable $B \cong A$ with nontrivial Π_1 automorphism.*

Proof. By Theorem 4.16, if A is η -like, then there is a Π_2 f with $\sum\{f(q) : q \in \mathbb{Q}\} \cong_{\text{rec}} A$. Now apply the Downey-Moses result. $\square$

Most of the above were proved using (variations of) the idea of a choice set. A *choice set* for a linear ordering is a set of elements, one from each block. To prove Theorem 5.19 we used a variation of this idea. If A has only finite blocks, then a *strong choice set* is one where we have the first element from each block. Now if f is a nontrivial automorphism of L then $\{x, f(x), f^2(x), \dots\}$ is an infinite ω or ω^* sequence in L . If L is computable and f is Π_1 this is a Σ_2 set. If g is a nontrivial Π_1 automorphism in B

of Theorem 5.19 and x has no predecessors, nor do $g(x), g^2(x), \dots$ and so $\{x, g(x), \dots\}$ is an infinite Σ_2 subset of a strong choice set.

The Downey-Moses [58] argument is to show that B can be constructed with no strong choice set with an infinite Σ_2 subset. The original use of Σ_2 choice sets was due to Lerman and Rosenstein [140] who showed that *there is a computable linear ordering of type $\zeta\eta$ with no Σ_2 dense subset*.

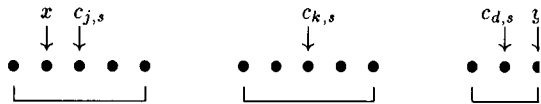
This argument was extended by Downey and Moses who showed

Theorem 5.20 (Downey and Moses [58]) *Every computable discrete linear ordering has a recursive copy with no Σ_2 choice set.*

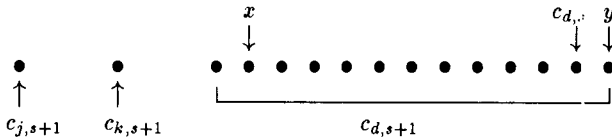
Sketch Proof. Again we use a variation of the Watnick construction. Let A have type $\zeta\tau$ so τ is a Π_2 order type. Let C have type τ with $C = \{c_0, c_1, \dots\}$. We know from Section 4 how to produce $B \cong A$ via Watnick's construction. The extra layer of requirements this time are to diagonalise against all the Σ_2 sets. If $\{R_e : e \in \omega\}$ lists all the r.e. binary relations, we define T_e via

$$n \in T_e \text{ iff } \neg(\exists_x^\infty) R_e(n, x).$$

Then $\{T_e\}_{e \in \omega}$ enumerates all Σ_2 sets. In the Tennenbaum variation of the Watnick construction we defeated ω and ω^* sequences. Here we defeat T_e 's by either making T_e too small or make $[x, y]$ finite for some $x, y \in T_e$. As in Theorem 5.2 there is no problem if x and y lie in (different) blocks with no blocks of higher priority between. As we did in Theorem 5.2 we might have



Then if $e < j, k, d$, we can amalgamate to get



Again we do have problems if $c_{j,s}$ and $c_{k,s}$ for instance had $j, k < e$. The point this time is that we will eventually get two we can combine if T_e is infinite. The fact that the T_e are Σ_2 and not computable creates some problems, but these can be overcome with the Π_2 method since we only need work with $\{e : |T_e| > e\}$ (which is a Π_2 set). So the outcome that we don't

need action to meet T_e is a Π_2 one and hence we can guess this on a tree of strategies. We don't give the details as they are quite tedious and contain no surprises. The reader is referred to Lerman and Rosenstein [140] or Downey and Moses [58]. $\square$

We remark that Theorem 5.20 can be proven also with orderings of type $\omega\tau$ or $\omega^*\tau$. Kierstead [121] defined an automorphism to be *fairly trivial* if for all x , $|[x, f(x)]| < \infty$. If a nontrivial automorphism is not fairly trivial, Kierstead called it *strongly nontrivial*. As a corollary to Theorem 5.20 we have:

Theorem 5.21 (Downey and Moses [58]) *Every computable discrete linear ordering has a recursive copy with no strongly nontrivial Π_1 self-embedding.*

Mathew Raw [187] independently proved the weaker result that there is a computable ordering of type $\zeta\eta$ with no strongly nontrivial Π_1 automorphism. He has also obtained the following related results:

- (i) *There is a computable linear ordering A (with uncountably many automorphisms) such that A has a Π_1 automorphism and which has no computable automorphism any other computable presentation of the ordering.*
- (ii) *The above problem (i) (for countably many automorphisms) is equivalent to: Is there a rigid Π_1 initial segment of a computable linear ordering which is not isomorphic to a computable linear ordering?*

The results of Downey and Moses on η -like linear orderings (Theorem 5.19) are obtained in a similar vein to the proof of Theorem 5.20, only there one tried to force an element to have an immediate predecessor to stop it being a Σ_2 subset of a *strong choice* set. Finally, we remark that Theorems 5.20 and 5.19 are quite sharp since Lerman and Rosenstein observed:

Observation 5.2 *Every computable linear ordering A has a Π_2 choice set.*

Proof. The Π_2 relation

$$(\forall x < a)(\forall y)(\exists z > y)[(x < z < a) \vee (a < z < y)]$$

defines a choice set. $\square$

There are quite a number of conjectures regarding Π_1 automorphisms. We list some below:

Conjecture 5.1

- (i) (Kierstead) *Every computable copy of type τ has a strongly nontrivial Π_1 automorphism iff τ has an interval of type η .*
- (ii) (Downey and Moses) *Every computable copy of τ has a fairly trivial Π_1 automorphism iff τ has elements x, y with (x, y) consisting only of block of type ζ .*
- (iii) (Downey and Moses [58]) *Every computable copy of τ has a strongly nontrivial Π_1 self embedding iff τ has a strongly η -like interval (i.e., (x, y) such that (x, y) is infinite and only has block of size $\leq k$).*
- (iv) (Downey and Moses [58]) *Every computable copy of τ has a fairly trivial Π_1 self-embedding iff τ has an interval (x, y) consisting only of blocks of type ω or ζ or only of blocks of type ω^* or ζ .*
- (v) (Kierstead [121]) *If a computable linear ordering has neither intervals of type ζ nor η then it has a copy with no nontrivial Π_1 -automorphism.*

Of course the basic question is to characterise the order types of Π_1 -rigid computable linear orderings.

Returning now to computable self-embeddings, we have the following that was surely noticed by all workers in the area. It was probably first noted by Watnick and Lerman.

Theorem 5.22 *If A is a strongly η -like computable linear ordering, then A has a nontrivial computable self-embedding.*

Proof. Let $A = \sum\{f(q) : q \in \mathbb{Q}\}$ with $f(q) \in \{1, \dots, n\}$ for all q . Knowing n we will know that when we see $(n+1)n$ many elements, the elements $1, n+1, 2n+1, \dots$ must all belong to different blocks. This clearly allows us to define the image of any block via a self-embedding. $\square$

Lerman, Watnick, Moses, and others have conjectured that Theorem 5.22 characterises the relevant order types.

Conjecture 5.2 (Folklore) *If τ is an ordertype containing computable linear orderings such that all computable linear orderings of type τ have a nontrivial recursive self-embedding, then τ has a strongly η -like interval.*

In support of Conjecture 5.2 it is possible to use the techniques of, for example, Theorem 5.20 to show that *if τ is η -like but has no strongly η -like intervals then a computable order type τ has a computable copy with no nontrivial recursive self-embedding.*

However, Conjecture 5.2 remains unresolved and, furthermore, the situation is quite complex as we will now see. In all of the constructions we have seen so far, counterexamples were obtained *uniformly* in the sense that there is a computable function f such that $W_{f(e)}$ is the relevant witness to W_e (if appropriate). For instance, in Remmel's theorem, Remmel actually shows that there is a computable function f such that if W_e is a computable linear ordering with infinitely many successivities, then $W_{f(e)}$ is isomorphic with W_e yet $W_{f(e)}$ is not computably isomorphic to W_e . It is unusual (but not unheard of) for constructions to be nonuniform. That is, for an example to exist, yet no f selecting an example to exist. The first example in classical recursion theory of this phenomenon was due to Lachlan [133]. We have seen it in the proof of our extension of Manaster's result where we don't know which happens i.e., either an ω or an ω^* sequence. We will now see that if Conjecture 5.2 is correct, there is no uniform procedure to prove it.

Theorem 5.23 (Downey) *There is no uniform procedure which, when given a computable linear ordering A produces a computable $B \cong A$ such that if A contains no strongly η -like interval then B has no nontrivial computable self-embedding.*

Proof. We build an infinite $(A, \leq)$ such that A has no strongly η -like intervals and we meet the requirements

R_e : If $W_e \cong A$ then $(W_e, \leq)$ has a nontrivial computable self-embedding.

The result will then follow by the use of the recursion theorem.

To meet R_e , we shall define a mapping $g_e = g$. If we let $z(e, i) = z(i)$ denote the i -th member of W_e in order of appearance, we must meet for all i .

$R_{e,i}$: $g(z(i))$ is defined.

To achieve (i) we make sure that either A is finite or, for almost all j , a_j ($=$ the j -th members of A) is part of an ω or ω^* or $\omega + \omega^*$ sequence. To do this we define $B(a_j, s)$ to denote the block containing a_j at stage s . We will ensure that $B(a_j, s) \subseteq B(a_j, s + 1)$ and hence we meet for almost all j

$$P_j : \lim_s |B(a_j, s)| = \infty.$$

To meet the requirements above, we proceed as follows.

We begin by enumerating a_0 . We win unless W_e enumerates $z(0)$. (If not, $|A| = 1$ and $|W_e| = 0$). It is now our goal to find a j with $z(j) < z(0)$, so to meet $R_{e,0}$ and define $g(z(0)) = z(j)$. To do this, we define $a_1 > a_0$. Now W_e can only respond in one of 3 ways. It can enumerate $z(1) < z(0)$, it can enumerate $z(1) > z(0)$ or it can do nothing. In the last case we win, in the second case we meet $R_{e,0}$. Only the first case creates any problems.

The idea is that we continue in this way (i.e., put $a_2 > a_1$, etc.) until some $z(j) > z(0)$ occurs. Note that if no such $z(j)$ occurs, we win as then $W_e \not\cong A$. The reason is that A is an ω -sequence yet W_e contains $z(0)$, a point with infinitely many points to the left of it.

Note that for the above strategy $|W_{e,s}| = |A_s|$ or $|W_{e,s}| = |A_s| - 1$ for all s . Of course, we control the enumeration of W_e to keep $|W_{e,s}| \leq |A_s|$ since if $|W_{e,s}| > |A_s|$ we can stop $A_s = A$ and force $W_e \not\cong A$. A typical situation is given below in Figure 6.

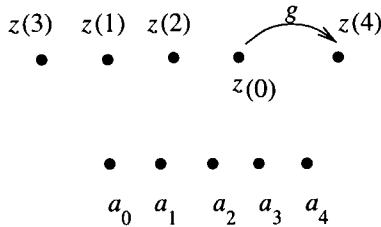


Figure 6.

At the end of this process (i.e., when we meet $R_{1,0}$) we then begin a 'clean up' process that helps to meet the P_j as we will see. The idea is simple. Before we meet any other $R_{e,i}$ we will ask that each *current* point of A is a member of block of at least one more member (i.e., if this happens at stage s). Thus we immediately give each member of A s successors. For example, if the s_1 function of Figure 6 happened at stage 10 we'd have (at stage 11) the situation of Figure 7 below:

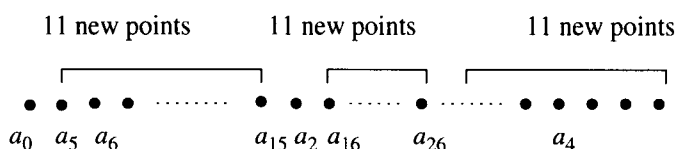


Figure 7.

The reason for this becomes clear later. Now we don't do anything until $|W_{e,t}| = |A_t|$ again at some $t > s$.

The problems that can occur are that we need to force points between others. A typical bad situation is described below:

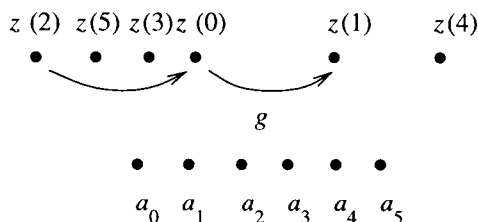


Figure 8.

This situation in Figure 8 is not quite accurate as there would always be more points in the actual construction.

We must try to get a point $z(j)$ with $z(0) < z(j) < z(1)$. *This is the heart of the construction.*

The basic action is to put a new point a_6 between a_3 and a_4 . W_e can respond in one of four ways:

- (i) no response (global win)
- (ii) $z(6) < z(0)$
- (iii) $z(6) > z(0)$
- (iv) $z(0) < z(6) < z(1)$ (win $R_{e,3}$).

In case (ii) we'd put a point between a_6 and a_4 and in (iii) we'd put a_7 between a_6 and a_3 . Continuing, assuming (i) and (iv) do not occur, there are three outcomes given by the order type of A we get

Outcomes 5.1

- (i) $\omega + k$: (iii) happens only finitely often and (ii) infinitely often.
- (ii) $k + \omega^*$: (iii) happens infinitely often and (ii) only finitely often.
- (iii) $\omega + \omega^*$: both (ii) and (iii) happen infinitely often.

In case (i) we win since the order type of W_e must be of the form $B + (k + 1)$; in (ii) it is $(k + 1) + B$ and in (iii) W_e must have $z(0)$ and $z(1)$, two points with infinitely many points each side of them. In any case $W_e \not\cong A$.

Of course if (ii) always occurs, then we get the desired self-embedding and A has the good order type in the sense that all elements are members of infinite blocks. The reason, of course, is that after the clean-up action, the place we need to add points to force W_e to respond can always be between two of the *added block points* and so achieve the goal of adding one point to each block. $\square$

The technique above can be applied to Conjecture 5.1. There seems no clear way to combine R_e strategies in the above for all R_e . One possible way is to use a $\mathbf{0}^{(3)}$ or $\mathbf{0}^{(4)}$ version of the ‘separator’ technique of Section 7, but at present this author can see no way to get this to work (and has tried very hard to do so).

One extension of Theorem 5.23 can be obtained if we control the complexity of the isomorphisms.

Theorem 5.24 (Downey) *There is a computable linear ordering A with no η -like interval such that for all computable B isomorphic to Z via a Δ_2 isomorphism, B has a nontrivial computable self-embedding.*

Proof. We meet the following modified R_e :

R_e : If $W_e \cong A$ via $\varphi_e(,)$ with $\varphi_e \Delta_2$ then W_e has
a nontrivial computable self-embedding.

Here $(W_e, \varphi_e)_{e \in \omega}$ is an enumeration of all r.e. subsets of $\mathbb{Q}$ and binary partial computable functions, so to meet R_e we need to ensure that either for some x , $\lim_s \varphi_e(x, s)$ does not exist, or $\varphi_e(x, s)$ is not total, or φ_e is not a homomorphism or W_e has a nontrivial computable self-embedding.

Using an idea along the lines of ‘separators’ we meet in Section 7, we divide A into infinitely many intervals $\{I_e\}_{e \in \omega}$. We play R_e on interval $I_e = [a_e, b_e]$. On this interval, we plan to perform the Theorem 5.23 strategy.

The idea is simple. We replace A of Theorem 5.23 by I_e and replace W_e by $W_e \cap [\varphi_e(a_e), \varphi_e(b_e)]$. So at stage s , we replace this by

$$\{z : z \in W_{e,s} \text{ and } \varphi_{e,s}(a_{e,t}) \leq z \leq \varphi_{e,s}(b_{e,t})\}.$$

If we reach a stage where $\varphi_e(a_e, t_1) = \varphi_e(a_e)$ and similarly b_e for all $t > t_1$, then we can win *between* $\varphi_e(a_e)$ and $\varphi_e(b_e)$ in exactly the same way as we did in Theorem 5.23. If t_1 fails to exist, we need only ensure that $[a_e, b_e]$ is not η -like and we win as $\varphi_e(a_e)$ has no value, so φ_e does not represent a Δ_2 function. $\square$

We remark that Theorem 5.24 is of some technical interest since most isomorphisms we have met, or are constructed for such problems, are Δ_2 . There are some examples of other ones. For instance in Section 7 we have an example where the witness is Π_2 . It is not clear if the above can be pushed to Δ_3 . Personally, I feel if it can be then there is an A such that all $B \cong A$ have nontrivial computable self-embedding, or there is a proof to the contrary where things are constructed at the Δ_3 level. One interesting possibility is that we could push Theorem 5.24 perhaps for all arithmetical isomorphisms.

6 Linear Extensions of Partial Orderings, Dilworth’s Theorem, and Effective Dimension

In this section, we will focus upon dimension and extension theory for orderings. We will be concerned with computable *partial* orderings, where $(A, \leq)$ is a computable partial ordering if A is identified with ω and given $x, y \in A$ we can decide which of $x < y$, $y < x$, $x = y$ or $x \mid y$ hold. An *extension* P of poset R is one where P and R have the same domains, and such that $x \leq y$ in P means $x \leq y$ in R . If R is a linear ordering, then we say that R is a *linear extension* of P . An old result of Szpilrajn [236] says that *any poset has a linear extension*. This leads to a theory of *dimension* for posets. The dimension of a poset P is the smallest ordinal n such that P can be expressed as the intersection of n linear extensions. This idea has led to an

interesting theory which contains such interesting theorems as Dilworth's decomposition theorem. We shall look at some effective aspects of this theory in this section.

It is not difficult to see the following.

Observation 6.1 (Folklore) *If P is a computable partial ordering, then P has a computable extension.*

Proof. The thing to note is that, with care, Szpilrajn's proof is effective. Let $P = \{a_0, a_1, \dots\}$. Gödel number all the pairs from P as $n_0, n_1, \dots$. At stage s , we attend to n_s . Let n_s be the Gödel number of (x, y) . If $x \leq y$ or $y \leq x$ (in R_s) to go stage $s + 1$. If $x : y$ declare $x \leq y$ (in R_{s+1}) and for all $a \leq x$ and $y \leq b$ (in R_s) declare $a \leq b$ (in R_{s+1}). Here $P_0 = P$ and we set $R = \cup_s R_s$.

Then R is the desired computable ordering. Note that R is computable as follows. Given m_1 and m_2 we wish to decide if $m_1 \leq m_2$ or $m_2 \leq m_1$. Let n_k be the Gödel number of (m_1, m_2) . Then in R_{k+1} we will know which of $m_i \leq m_{i+1}$ holds, since either we have $m_i \leq m_{i+1}$ in R_k or we will attend to (m_1, m_2) in R_{k+1} . $\square$

Much of the classical research about linear extensions has focused upon *preservation*. That is, if R has property Q does there exist an extension of R with property Q ? (One could also look at the Q -dimension of R , namely the least n , if any, such that R can be written as the intersection of n orderings all with property Q . I am not aware of much work in this area.) For instance, we can show that if R lacks an ω^* sequence (i.e., it is well founded) then R has a well founded linear extension (Bonnet [23]). In fact, Bonnet (see Bonnet & Pouzet [25]) completely classified the countable linear order types τ such that if R is countable and τ -free then R has a τ -free linear extension.

Question 6.1 *Is there an effective analogue of Bonnet's result?*

Rosenstein, Kierstead and Statman have investigated the effective content of the earlier result of Bonnet.

Theorem 6.1 (Rosenstein, Kierstead (see Rosenstein [209])) *If P is a well founded computable poset, then P has a well founded computable extension.*

Rosenstein [209] remarks that the standard proof fails to prove Theorem 6.1. The standard proof lets A_0 be the set of R -minimal elements and A_j to be the minimal elements of $A - \cup_{i < j} A_i$. Then one well orders the A_j 's and puts the rest together. This proof would give a Π_2 extension.

Proof of Theorem 6.1. The idea is to modify the proof of Observation 6.1 (essentially). Suppose, at stage s , we have P an ordering of $\{a_0, \dots, a_{s-1}\}$ extending P on $\{a_0, \dots, a_{s-1}\}$. Suppose the P -ordering is $a_{i_0} < \dots < a_{i_{s-1}}$. Find the R_s -smallest a_{i_t} such that $a_s <_P a_{i_t}$ and set $a_{i_{t-1}} < a_s < a_{i_t}$. If a_{i_t} does not exist then put $a_s > a_{s-1}$. Then P is a linear extension, so we need only prove that P is well founded.

Suppose $\{a_{j_0} > a_{j_1} > \dots\}$ is an ω^* sequence in R . By Ramsey's theorem, we can assume that for all $k \neq m$, $a_{j_k} \mid a_{j_m}$ in P since we know P is well founded. We can also suppose that $j_k < j_{k+1}$ for all k .

Since $a_{j_k} <_R a_{j_0}$ but $a_{j_k} \mid_P a_{j_0}$ it follows that there is some c_k with $a_{j_k} <_P c_k <_R a_{j_0}$. Note that as $a_{j_k} \not<_P a_{j_0}$, $c_k \not<_P a_{j_0}$. It follows that if $c_k = a_{i_t}$ then $i_t < j_0$, by construction. Choose i_t least. It follows that there is some *least* d_1 with $a_{j_i} <_P d_1 <_R a_{j_0}$ for infinitely many i and hence we can thin the sequence to get, without loss of generality, $a_{j_0} <_R d_1 <_P a_{j_i}$ for all i . We now continue inductively to get a sequence $a_{j_0} > a_{j_1} > \dots$ with $a_{j_i} <_R d_i <_P a_{j_{i+1}}$. Using the minimality argument above, it is easy to see that $d_1 <_P d_2 <_P \dots$ giving a contradiction. $\square$

A couple of comments on this proof. First, in Rosenstein [209] it was noted that the argument gives a (new) proof that if α is an infinite cardinal and $\alpha \not\leq |P|$, then there is a linear extension R of P such that $\alpha \not\leq |R|$. Secondly, the argument is very reminiscent of the 'minimal bad sequence' arguments used in well quasi-ordering and better-quasi-ordering proofs. This suggests that there may be some connections between computability and bqo theory.

It was noted by Rosenstein [209] that a key obstacle in solving Question 6.1 is dealing with scattered types.

Question 6.2 (Rosenstein [209]) *If R is scattered and computable, does R have a scattered computable extension?*

The classical version is given in Bonnet and Pouzet [25] and Galvin and McKenzie (unpublished), and is part of Bonnet's classification.

We remark that it is not totally clear what the correct analogue for Question 6.1 ought to be. As is often the case, a classical theorem can have multiple effectivisations. For instance, we can ask if being ‘*computably well founded*’ is preserved. Here we say that an ordering A is *computably well founded* (resp. *computably scattered*) if it has no *computable* ω^* sequence (resp. does not *computably embed* η).

Theorem 6.2 (Rosenstein and Statman (in Rosenstein [209])) *There is a computably well founded computable partial ordering with no computably well founded computable linear extension.*

Proof. Let T be the computable collection of orderings σ such that $T = \cup_s T_s$ defined as follows:

$$\sigma \notin T_s \quad \text{iff} \quad (\exists t)(\exists \gamma)(\gamma \in W_{e,t} \text{ and } t < s \text{ and } \ell h(\gamma) > 2e \text{ and } \gamma \subseteq \sigma).$$

Then T is a computable tree and is infinite by König’s Lemma. Define the usual partial ordering on T via $\lambda, \gamma > \sigma$ iff $\gamma \subseteq \sigma$. Then $>$ is a computably well founded partial ordering on T since if Q is a branch of T then $W_e \not\subseteq T$ for any e .

Let B be a computable linear extension of T . Define a computable ω^* -sequence Q as follows. At stage s , we choose b_s from level s . At stage 0 let $b_0 = \lambda$. At stage s let P_s denote the collection of immediate predecessors of $Q_{s-1} = \{b_0, \dots, b_{s-1}\}$ met already in Q_{s-1} . Let a_s be the B -largest element of P_s . Then Q is clearly computable. The claim is that $a_{s+1} <_B a_s$ for all s . If $a_s <_B a_{s+1}$ then $a_{s+1} \notin Q_t$ for all $t < s$. But $a_{s+1} \in Q_s$, and hence a_{s+1} is an immediate predecessor of a_s , so $a_s <_A a_{s+1}$ and we are done. $\square$

Question 6.3 (Rosenstein [209]) *What is the situation with computably scattered partial orderings?*

Rosenstein [208, 209] observed that one can resurrect effective versions using the arithmetical hierarchy.

Theorem 6.3 (Rosenstein [209]) *Let R be a computable partial ordering. Then*

- (i) *If R is computably well-founded, it has a Δ_2 computable well founded linear extension.*

- (ii) If R is computably scattered, it has a Δ_2 computably scattered linear extension.

Proof. (i), for example. One constructs an extension of R ensuring that if $W_e = \{a_0, a_1, \dots\}$ is infinite then for some $i < j$ that $a_j <_B a_i$. Using a priority argument, this needs only a Δ_2 oracle. $\square$

Question 6.4 Is Δ_2 the best we can do here? Can it be replaced by Π_1 ?

One can also ask in Question 6.4 what degrees such extensions might have. There are many other preservation properties of linear extensions for which one might ask the effective content (see Bonnet and Pouzet [25]). Another topic of interest here is:

Question 6.5 Classify the computable partial orderings with exactly one computable linear extension.

Recently Slaman and Woodin [225] considered the question of finding necessary and sufficient conditions for a poset to have a dense linear extension without endpoints. (This was a question of Loś.) Slaman and Woodin demonstrated that the problem has no simple solution by proving:

Theorem 6.4 (Slaman and Woodin [225])

$$\left\{ e \mid W_e \text{ is a computable partial ordering of } \omega \text{ and} \right. \\ \left. \text{there is an endpoint free dense linear extension of } W_e \right\}$$

is Σ_1^1 complete.

The proof of the above would take us beyond the scope of the present paper, and involves coding a new Π_1^1 set which Slaman and Woodin call “almost perfect well founded trees”.

The next topic we turn to is the topic of dimension. The *computable dimension* of a poset A is the least d such that A can be *computably embedded* into $\mathbb{Q}^d$. This corresponds to the classical notion of dimension introduced by Dushnik and Miller [61] since all countable linear orderings are embeddable in $\mathbb{Q}$. As we might expect, the Dushnik-Miller theorem has a number of effectivisations, even for *two dimensional partial orderings* (2-dpo’s). The

2-dpo's have been studied quite extensively (see e.g., Kelly and Trotter [109], Trotter and Moore [238], Baker, Fishburn and Roberts [14]) and as the 'next simplest' poset beyond a linear ordering, their combinatorics and computable combinatorics are of some interest. Manaster and Rosenstein [149] initiated the study of computable 2-dpo's. The notions of computable 2-dpo and computable poset with computable dimension two are quite different as we will see.

Theorem 6.5 (Manaster and Rosenstein [149]) *There is a computable 2-dpo that does not have computable dimension 2.*

In fact, Manaster and Rosenstein improved Theorem 6.5 to show:

Theorem 6.6 (Manaster and Rosenstein [149]) *There is a computable 2-dpo B that is not isomorphic to a computable subordering of $\mathbb{Q}$.*

Lemma 6.7 (Manaster and Rosenstein [149]) *There is a sequence of finite 2-dpo's, none of which is embeddable in the other.*

Proof. Let $n \geq 4$ and let D_n be the ordering of the $2n+1$ points $d_1, \dots, d_n, q_1, \dots, q_{n-1}, p_1, p_2$ ordered by the transitive closure of:

$$q_i < d_i \text{ and } q_i < d_{i+1} \text{ for } 1 \leq i < n$$

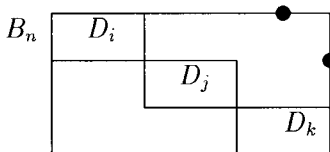
$$d_i < p_1 \text{ for } 1 \leq i < n \text{ and } d_j < p_2 \text{ for } 1 < j \leq n.$$

It is routine to show that if D_m is embeddable into D_n then $m = n$. $\square$

Proof of Theorem 6.6. The argument is quite elegant. The plane is divided into a sequence of disjoint boxes $B_1, B_2, \dots$. Into B_n we put all of the points of D_{3n+i} for $i \in \{1, 2, 3\}$ where $\{D_j : j > 4\}$ lists a computable set of 2-dpo's of Lemma 6.7. Use the even numbers to do the Gödelization. B_i may contain two additional points. The only new relations are determined by the construction and that if $i < j$ and $x \in B_i$ and $y \in B_j$ then $x <_B y$.

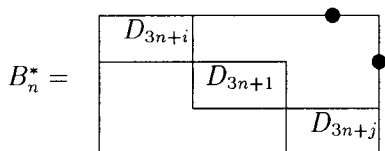
Let W and V be two computably enumerable *computably inseparable* sets. That is $W \cap V = \emptyset$ and there is no computable set R with $W \subset R$ and $R \cap V = \emptyset$. (These are easy to construct. At stages s , if $W_s \cap W_{e,s} = \emptyset$ and there is a $z \in W_{e,s}$, $z \notin V_s$ and $z > 4e$, put z into $W_{s+1} - W_s$ (similarly V_s) and let $W = \cup_s W_s$, $V = \cup_s V_s$).

The picture of B_n is:

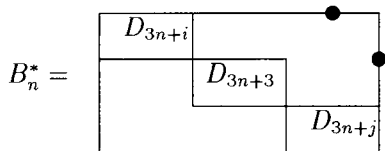


At stage s , if n occurs in W_s or V_s the first two odd numbers not used are placed in B_n as marked. In detail, if $n \in W_s$, the first point is placed above all points of D_{3n+1} and D_{3n+2} but made incompatible with D_{3n+3} and the second placed above all of D_{3n+1} and D_{3n+3} but incompatible with D_{3n+2} . If n occurs in V , the first of the two points is placed above all the points of D_{3n+2} and D_{3n+3} , but incomparable with D_{3n+1} whilst the second above D_{3n+1} and D_{3n+3} but incompatible with D_{3n+2} . Clearly R so construed is computable. Moreover it is a 2-dpo via the representation $B_0^* < B_1^* < \dots$ with B_n^* described as follows.

$$B_n^* = D_{3n+i} < D_{3n+j} < D_{3n+k} \quad \text{if } n \notin W \cup V, i \neq j \neq k \in \{1, 2, 3\}$$



with $i \neq j$ and
 $i, j \in \{2, 3\}$ if $n \in W$



with $i \neq j$ and
 $i, j \in \{1, 2\}$ if $n \in V$

Suppose now that B is isomorphic to a computable subset A of the plane. We claim that we can computably separate W and V by a set R . Search B until subsets D'_{3n+1} , D'_{3n+2} , D'_{3n+3} are found with $D'_{3n+i} \cong D_{3n+i}$. These clearly must exist. Moreover, by the construction of B , we know that they must be in the orientation given by the B_n^* 's. Let P_i be a point of D'_{3n+i} . Determine which is left uppermost and which is right lowermost, and let P_t be the last one. If $t = 1$ put $n \in R$ otherwise put $n \notin R$. R is then a computable separator of W and V . $\square$

A similar argument will show:

Theorem 6.8 (Manaster and Rosenstein [149]) *There is a computably enumerable subordering of $\mathbb{Q}^2$ not isomorphic to a computable one.*

(This should be compared with the corresponding situation for linear orderings.) While $\mathbb{Q}^2$ is not the universal object for 2-dpo's, Manaster and Rosenstein did manage to find one.

Theorem 6.9 (Manaster and Rosenstein [149]) *There is a computable 2-dpo, U such that if R is a computable 2-dpo then R is isomorphic to a computable subset of U .*

Sketch Proof. Use an infinite set of boxes B_i as before, now each containing an infinite computable set $R_e = \{b_0^e, b_1^e, \dots\}$. We split B_e into two portions, one for 'padding' and one to embed L_e , the e -th 2-dpo (which may be finite). We declare all the points in the lower portion to be less than those in the upper. At stage s we put b_e^s as part of an ω -sequence in the lower portion if no new element occurs in $L_{e,s}$. If a new element occurs, put it in the appropriate place in the upper portion to keep it isomorphic with $L_{e,s}$. $\square$

Manaster and Remmel [147] made similar investigations into *dense* computable 2-dpo's. Since our concern is mainly with linear orderings we shall be rather brief here. By a dense 2-dpo we mean one that is isomorphic to a topologically dense subset of $\mathbb{Q}^2$. Some special dense 2-dpo's are D_2 , a computable subset of $\mathbb{Q}^2$ with no 2 points collinear (this characterises D_2 up to isomorphism (see Manaster and Remmel [146]), and $D_2^{m,n}$ where this is the result of augmenting D_2 by m horizontal and n vertical lines (in some computable fashion). D_2 is like a universal structure for computable dense 2-dpo's.

The density of a 2-dpo adds quite a lot of room for manoeuvring. For instance, in contrast to Theorem 6.6 we have via a Cantor back-and-forth argument:

Theorem 6.10 (Manaster and Rosenstein [149]) *The following are equivalent for a dense 2-dpo P*

- (i) *P is computably isomorphic to a computably presented 2-dpo.*
- (ii) *P is computably isomorphic to a computable subset of D_2 .*
- (iii) *P is isomorphic to an computably enumerable subset of D_2 .*

Proof. Evidently (ii) $\Rightarrow$ (iii) $\Rightarrow$ (i), so we need only show that (i) $\Rightarrow$ (ii).

In Manaster and Remmel [147] it is shown that the coordinate orderings $\leq_1, \leq_2$ for *dense* 2-dpo's are definable in the theory of dense 2-dpo's via constants i and j which are interpreted as incompatible, $i \mid_P j$ and agreeing $i <_1 j <_2 i$. So we can view $r <_1 s$ meaning r is left of s and $r <_2 s$ means r is below s . It is not necessary that $<_1$ and $<_2$ be computable just because $<$ is, but Manaster and Remmel show they are computably enumerable as follows. Define $x <_1^* y$ to mean $x <_1 y \vee (x =_1 y \text{ and } x <_2 y)$ and similarly $x <_2^* y$ to mean $x <_2 y \vee (x =_2 y \text{ and } x <_1 y)$. Then one can show that $<_1^*$ and $<_2^*$ are computable using the fact that $<$ is computable as $<_1^*$ and $<_2^*$ are both computable, we can define a computable f with $x <_k^* y$ iff $f(x) <_k f(y)$ in D_2 for $k = 1, 2$ via the density of D_2 . This allows us to show that $f(P)$ (in D_2) is computably enumerable and hence $f(P)$ is computable. $\square$

This proof also shows that a dense 2-dpo P is isomorphic to a computable subset of D_2 iff P is isomorphic to a computably enumerable subset of D_2 . The technique of proof would also be able to prove that D is isomorphic to a Π_n subset of D_2 iff it is isomorphic to a Σ_{n+1} subset of D_2 (as we saw with the linear orderings.)

It is not true that an computably enumerable P is computably isomorphic to a computable subset of D_2 . Actually, here we need some care with what we mean by D_2 . Whilst D_2 is characterised up to isomorphism, it is not so up to *computable* isomorphism. The key notion is that of a *decidable* presentation. Recall that for a computable model $\mathbf{A}$ we say that a presentation is decidable if there is a computable procedure R such that given any first order formula φ in the language of $\mathbf{A}$, R can decide if φ is true of $\mathbf{A}$ or not. Using a wait and see argument, Manaster and Remmel showed the following.

Theorem 6.11 (Manaster and Remmel [147]) *There is an computably enumerable presentation of D_2 not computably isomorphic to a computable subset of a decidable presentation of D_2 .*

Sketch Proof. We build a computable subset D of $\mathbb{Q}_2$ that is isomorphic with D_2 . Let B denote the decidable presentation of D_2 . We meet for $e \in \omega$

R_e : φ_e is not a computable embedding of D into B .

To aid the construction, one builds a Δ_2 isomorphic $f = \lim_s f_s$ from D to B , in a fashion similar to many of our earlier arguments.

The basic idea is essentially the following. If R_e is not yet (declared) satisfied, we wait till there exist a_e and b_e with

- (i) $f^s(a_e) = \langle 2e, 2e + 1 \rangle$, $f^s(b_e) = \langle 2e + 1, 2e \rangle$,
- (ii) $\varphi_e^s(a_e) \downarrow$ and $\varphi_e^s(b_e) \downarrow$ (and φ_e^s is currently consistent on its domain);
and
- (iii) $\varphi_e^s(a_e) \upharpoonright_B \varphi_e^s(b_e)$.

The action is to then make $a_e \leq_D b_e$ by moving the f_{s+1} partial isomorphism around, using the density of D_2 to do this without disturbing our earlier work for any other R_j . $\square$

The ideas above are elaborated and used in quite a number of the proofs of Manaster and Remmel [147], giving results such as:

Theorem 6.12 (Manaster and Remmel [147])

- (i) $\mathbb{Q}^2$ is decidablely categorical (i.e., any two decidable presentations of $\mathbb{Q}^2$ are computably isomorphic) and so is D_2 . Furthermore, if P is a decidablely presented dense 2-dpo, then P is computably isomorphic to a computable subset of a decidable presentation of $\mathbb{Q}^2$.
- (ii) $\mathbb{Q}^2$ is not computably categorical but D_2 is (so all computable presentations of D_2 are decidable ones).
- (iii) $D_2^{1,0}$ is decidablely categorical but not computably categorical.
- (iv) For any p and q with $p + q > 0$ there exists a computably rigid computable presentation of $D_2^{p,q}$. (Note that $D_2^{p,q}$ has $2^{\aleph_0}$ classical automorphisms; and D_2 is not computably rigid (via Cantor)).

We remark that the proof of (iv) requires a much more elaborate finite injury priority argument. The positive results above are proven by back-and-forth arguments. For more on dense 2-dpo's the reader is referred to Manaster and Remmel [147, 146].

Remmel [190] and Metakides and Remmel [161] have some other work associated with dense 2-dpo's. Again this is rather outside the scope of this paper, so we restrict ourselves to a few brief comments with respect to the flavour of this work.

The computably enumerable sets form a lattice $\mathcal{E}$ which has been extensively studied. The main approach is to study the quotient lattice $\mathcal{E}^*$, that is, $\mathcal{E}/\equiv^*$, where

$$A \equiv^* B \quad \text{iff} \quad |(A - B) \cup (B - A)| < \infty.$$

One seeks to understand the relationships between the structure of $\mathcal{E}$ and the computably enumerable degrees. On the other hand, when we look for the analogue of $\mathcal{E}^*$ in other computable structures, $\equiv^*$ is often not the correct notion. For instance, in the lattice of subspaces, the analogue of $\equiv^*$ is “ $\equiv$ modulo a finite dimensional piece”.

Metakides and Remmel [161] introduced a notion (*in a general position* (i.g.p.)) — corresponding to the notion of infinite/cofinite in $\mathcal{E}$ which pertained to a wide class of decidable models M including, for example, $\mathbb{Q}$ and $D_2^{m,n}$ (and $\mathbb{N}$). In this paper and subsequently Remmel [190] a number of satisfying extensions of results for $\mathcal{E}$ were obtained. We give only one example. A set A in the model M is said to be i.g.p. if for every infinite set $X \subseteq \mathbb{N}$ definable from finitely many parameters of M , both $X \cap A$ and $X - A$ are infinite. A set $A \subseteq M$ is called *cohesive* (i.g.p.) if A is i.g.p. and for any computably enumerable set $W \subseteq M$ either $W \cap A$ or $A - W$ is not i.g.p. (in $\mathcal{E}$ a set A is cohesive if for all computably enumerable sets W either $|W \cap A| < \infty$ or $|A - W| < \infty$).

Theorem 6.13 (Metakides and Remmel [161]) *If M is appropriate, as above, then M has a co-computably enumerable cohesive set i.g.p. (this generalises the well-known result of Friedberg [73] for $\mathcal{E}$).*

Returning now to the main line of this section, we have seen that computable dimension and dimension are quite different. A famous classical theorem of Dilworth [47] asserts that if D is a poset with finite width n (i.e., the largest antichain has size n) then D can be written as the union of n disjoint chains (and, indeed, the dimension of D is thus $\leq n$).

Following Kierstead, McNulty and Trotter [122], we call the least n such that D can be written as the union of n disjoint computable chains the *computable chain covering number*. The exploration of the connections between these notions leads to some fascinating and rather complex combinatorics. Kierstead et al [122] improved the Manaster-Rosenstein result Theorem 6.5 to show:

Theorem 6.14 (Kierstead, McNulty and Trotter [122]) *There is a computable ordered set A of width 3 with computable chain covering number 4 and no computable (finite) dimension.*

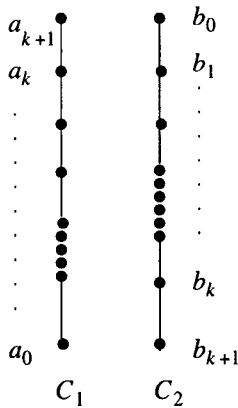
One of the key notions we will need is that of a *computable realizer*. We say that $(L_1, \dots, L_n)$ is a (computable) n -realizer for a computable P if $P = L_1 \cap \dots \cap L_n$ with the L_i all computable linear orderings.

Proof. To prove Theorem 6.14 we must build a computable ordered set A of width 3 and computable chain cover number 4 to meet

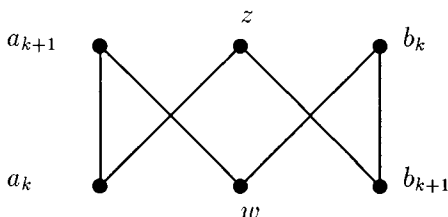
$$R_e : A \neq \bigcap_{i \in D_e} L_i$$

where L_i denotes the i -th computably enumerable linear ordering and D_e the e -th canonical finite set. That is, we diagonalise over all computable realizers.

For a single R_e we proceed as follows. We begin to build chains C_1 and C_2 as described in the diagram below.



Following Kierstead, McNulty and Trotter [122] define pairs (x, y) and (p, q) to be *pointlike* (relative to D_e) if $x < y$ and $p < q$ (in A) and for all $i \in D_e$ either $q <_i x$ or $y <_i p$ where $<_i$ denotes the order relation on L_i . When we add a_{k+1} and b_{k+2} we wait to see if the pair $(a_k, a_{k+1}), (b_k, b_{k+1})$ is pointlike before we define a_{k+2} and b_{k+2} . The goal is to find a pointlike pair $(a_k, a_{k+1}), (b_k, b_{k+1})$. If we find such k we meet R_e as follows. We enumerate $z \in C_3$ and $w \in C_4$ as follows:



Now suppose in some L_i we have $z <_i w$. then as the pairs are pointlike, either $b_k <_i a_k$ or $b_{k+1} <_i a_{k+1}$. If, for instance, $b_k <_i a_k$ we have $z <_i w <_i b_k <_i a_k <_i z$. From this it follows that in any L_i we must have $w <_i z$ and hence $A \neq \bigcap_{i \in D_e} L_i$.

It suffices to show for a single R_e that a pointlike pair must exist. After the pair (a_j, b_j) is added, let m_j denote the number of L_i with $i \in D_e$ such that $a_j > b_j$. Then assuming we are consistent with $A = \bigcap_{i \in D_e} L_i$ we must have $1 \leq m_j \leq |D_e| - 1$ for all j . This follows since $a_j \mid b_j$ so $a_j > b_i$ for some i , but not all i . (Of course if this fails, we simply stop and win R_e outright.) We know that $a_{j+1} > a_j$ and $b_j > b_{j+1}$ in A . Hence if $a_j > b_j$, we have $a_{j+1} > b_{j+1}$. Thus $m_j \leq m_{j+1}$ and hence, by the pigeonhole principle, for some $k < |D_e|$ we have $m_k = m_{k+1}$. For this k , $(a_k, a_{k+1}), (b_k, b_{k+1})$ is pointlike.

To complete the proof, it suffices to say how to combine the R_e 's. But this is easy. Split the construction into infinitely many levels with R_e working (locally) at level e (which can be thought of as a box in which we meet R_e). We declare the boxes ordered in A via box $e < \text{box } f$ if $e < f$. Finally note that $C_1 \cup C_2 \cup C_3 \cup C_4 = A$. $\square$

At this stage, we remark that the results of Kierstead et al [122] and Kierstead [117, 120] all had proofs cast in the language of computable game theory. Here we choose not to do this since we would like to give an alternative view. Most (all) of the constructions of recursion theory or computable combinatorics can be recast in this way, but it is not clear if such an approach yields greater insight. There is some evidence each way. We remark that Theorem 6.14 leads to a fundamental question.

Question 6.6 What are the precise relationships between $W(A)$, the width of A , $C(A)$, the computable covering number of A and $d(A)$, the computable dimension of A ?

The reader should recall that in Manaster and Rosenstein [149] produced a computable 2-dpo that was not even isomorphic to a computable subset of $\mathbb{Q}^2$. Theorem 6.14 leaves open the following question.

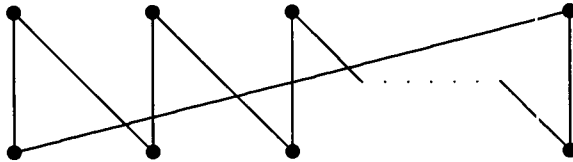
Question 6.7 *Is there a computable A of finite width such that A is not isomorphic to a computable subset of $\mathbb{Q}^n$ for any n ? What about computable ordinals n ?*

Also,

Questions 6.8

- (i) *For a given n , what values can $d(A)$ assume for A of computable width n ? Is each $m \geq n$ possible?*
- (ii) *What happens if A is dense?*

Kierstead et al [122] observed that the poset of Theorem 6.14 has many induced suborderings that are 3-crowns. Here an n -crown is a collection of $2n$ points forming the following Hasse diagram.



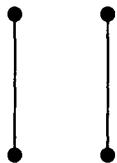
Using an elaboration of the idea of pointlike pairs of Theorem 6.14, these authors showed:

Theorem 6.15 (Kierstead, McNulty and Trotter [122]) *Let $c > 1$. There exists a computable crown-free poset A with $c(A) \leq c$ such that $d(A) \geq c \binom{c-1}{t}$ where $t = \lfloor (c-1)/2 \rfloor$.*

Kierstead [120] points out that it is not clear if one needs to exclude only 3-crowns in Theorem 6.15.

Question 6.9 *Is there any condition we can put on A to ensure, for instance, $d(A) \leq w(A)$ or $d(A) \leq c(A)$ or $c(A) = w(A)$? Also, if $c(A) = w(A)$, is it true that $d(A) < \infty$?*

One class of orderings that has attracted some attention are *interval orderings* since these form a relatively well understood subclass of the crown-free orderings. For the reader unfamiliar with the notion, one characterisation is that P is an interval ordering iff P has no subordering of the type given below



The name actually comes from the manner in which such orderings can be represented. In the negative direction, Hopkins improved the results of Kierstead et al [122] for interval orders:

Theorem 6.16 (Hopkins [100]) *There are computable interval orders of width w and computable dimension $\lceil \frac{4}{3}w \rceil$.*

Sketch Proof. Let A be an interval ordering. Define, for $x \in A$,

- (i) $U(x) = \{y : x < y\}$ and $D(x) = \{y : y < x\}$, and
- (ii) $U_*(x) = \{y : U(x) \subset U(y) \text{ and } x \neq y\}$ and
- (iii) $D^*(x) = \{y : D(x) \subset D(y) \text{ and } y \neq x\}$.

We call an element down (up) in a linear extension L if x is less (greater) than all elements of $D^*(x)$ ($U_*(x)$) in L . Given w , we must build A of width w to meet

$$R_e : \text{ If } t = \lceil \frac{4}{3}w \rceil - 1 \text{ and } |D_e| = t, \text{ then } A \neq \bigcap_{i \in D_e} L_i.$$

For the discussion of a single R_e , we will simply let $D_e = \{1, \dots, t\}$.

The initial play is to build $\frac{1}{2}w$ element antichains A and B with $B < A$. Now we wait until $L_1 \cap \dots \cap L_t$ covers these elements. Define $f(m)$ ($g(m)$) to be the unique element of $A(B)$ that is down (up) in L_m . Now we appeal to the following combinatorial hashing lemma.

Lemma 6.17 (Trotter and Munroe [239]) *If f and g are functions from $\{1, \dots, t\}$ to $\{1, \dots, w\}$, where $t = \lceil \frac{4}{3}w \rceil - 1$, then there exist $i, j \in \{1, \dots, w\}$ such that $f(m) = i$ and $g(n) = j$ imply $m = n$.*

By Lemma 6.17 there are $a \in A$, $b \in B$ and m with a only down in L_m and b only up in L_m . Then we win by putting a new point q below all elements of A except a and above all elements of b except b . Then in L_m it must be that $c < a$ and $b < c$, this is a contradiction.

The strategies again fit together in the obvious way. $\square$

Before we look at any further negative results, we will now examine a rich collection of very surprising positive results. We begin with Kierstead's effective version of Dilworth's theorem.

Theorem 6.18 (Kierstead [117]) *A computable ordered set A of width m has computable chain cover number $\leq (5^m - 1)/4$.*

Proof. The argument is quite complex and is described in great detail in [117]. Another account is given in Kierstead [120]. We will only sketch the details following Kierstead [120]. The argument is proven by induction, and the strategy is given by recursion.

Let $n = (5^m - 1)/4$, and suppose the result for all computable orderings of width $\leq m$. Suppose A has width $m + 1$. Let $k = (5^{m+1} - 1)/4 = 1 + 5(5^m - 1)/4$. We need to construct chains $C_1, \dots, C_k$. These we label as $B, C_{i,j}$, $1 \leq i \leq n$, $1 \leq j \leq 5$. At stage s , the idea is to try to put s into B . If we cannot do this, we put s into a set Q . On this set Q there is a width m computable order R extending A 's orderings O which we define. This ordering R has certain special properties. We appeal to the induction hypothesis to use the strategy for width m orderings to put s into one of the n R -chains $D_1, \dots, D_n$. This is okay by the uniformities of the proof. Finally the special properties of R will ensure that if s is put into D_j then s is put in one of $C_{j,1}, \dots, C_{j,5}$ which cover D_j .

Property 6.1 *The exact properties we need for R are that it extends A , and (at stage s , dropping the s 's).*

(i) (Q, B) is a partition of A with

- (a) B is a maximal O -chain
- (b) $O|Q \subseteq R|Q$
- (c) $D_1, \dots, D_m$ covers Q

- (d) if $x <_R z$ and $x|b|z$ in O , for some $b \in B$ enumerated before the latter of x and z , we have $x < z$ in O . Furthermore, if $x < y < z$ in R then $y|b$ in O .
- (ii) $(Q, \sim)$ is an equivalence relation such that
- (a) $x \sim y$ implies $(x < y \text{ or } y < x \text{ in } O)$.
 - (b) The equivalence classes of $(A, \sim)|D_j$ are convex in R for all j .
(That is, $x, y, z \in D_j$, $x \sim z$ and $x < y < z$ implies $x \sim y$ and $y \sim z$.)
- (iii) For each j , $C_{j,1}, \dots, C_{j,5}$ is a chain cover of D_j in O such that if $x \sim y$ then $x \in C_{j,k}$ iff $y \in C_{j,k}$.

To ensure that these properties hold at each stage the induction goes through, the play is as follows.

STEP 1:

Given $Q_s, B_s, R_s, D_1^s, \dots, D_m^s$, i.e., $Q, B, R, D_1, \dots, D_m$ at stage s , we need to place s . If $B_s \cup \{s\}$ is a chain, put s into B_{s+1} . Otherwise put s into Q_{s+1} . Now let $x < s$ ($s < x$) in R_{s+1} iff one (or more) of the following holds:

- (i) $x < s$ ($s < x$) in (A, O)
- (ii) For all $b, c \in B$ if $b|x$ and $c|s$ in O then $b < c$ ($c < b$) in O .
- (iii) $(\exists a \in Q)(x < a$ ($a < x$) in R_s and (i) and (ii) above hold with x replaced by a).

Now use the inductive strategy to put s into one D_j . (Note the verification requires us to show that the width of Q_{s+1} remains at m).

STEP 2:

Next we need to construct $\sim_{s+1}$, $C_{i,j}^{s+1}$ for all $1 \leq i \leq n$, $1 \leq j \leq 5$. Assume $s \in D_j^{s+1}$. Let s^- (s^+) be the largest (smallest) element D_j^{s+1} less than (greater than) s in R_{s+1} . Extend $\sim_s$ to $\sim_{s+1}$ via:

- (i) s is in the equivalence class of s^- if there exists $b \in B_{s+1}$ with $s|b|s^-$ in $(A, 0)$.

- (ii) s is in the equivalence class of s^+ if (i) does not pertain *and* there is $b \in B_{s+1}$ with $s \mid b \mid s^+$ in $(A, 0)$.
- (iii) If neither (i) nor (ii) pertain, then s is in a new equivalence class.

If $s \sim_{s+1} x$ and $x \in C_{j,k}^s$, put s into $C_{j,k}^{s+1}$. Otherwise put s into $C_{j,e}^{s+1}$ where e is chosen so that neither of the two equivalence classes over s in $(Q_{s+1}, R_{s+1}) \upharpoonright D_j^{s+1}$ nor those immediately below s in $(Q_{s+1}, R_{s+1}) \upharpoonright D_j^{s+1}$ are contained in $C_{j,e}$.

To complete the proof one must give an exact analysis of the construction to show that the properties 6.1 are indeed preserved by the construction. The details are not difficult, but don't really give any further insight save to note that the conditions 6.1 specifically chosen to allow the induction on stage number to proceed. We refer the reader to, in particular, Kierstead [120, page 87] where these are all spelled out. $\square$

Szemerédi gave a lower bound for Theorem 6.18 via the following which examines the effective content of Mirsky's [163] dual to Dilworth's theorem.

Theorem 6.19

- (i) (Schmerl, see Kierstead [120]) *Every computable ordering of finite height h (i.e., the longest chains have size h) can be covered by $\binom{h+1}{2}$ computable antichains.*
- (ii) (Schmerl, see Kierstead [120]) *For all h there is a computable ordering of height h that cannot be covered by fewer than $\binom{h+1}{2}$ computable chains.*
- (iii) (Szemerédi, see Kierstead [120]) *In (ii) we can choose the counterexamples to have computable dimension 2.*

Proof.

- (i) Let P have height h and let $n = \binom{h+1}{2}$. We construct antichains $A_1, \dots, A_n$ to cover P . Label the antichains as $A_{d,u}$, $0 \leq d+u \leq h-1$. At stage s , put s into $A_{d,u}^{s+1}$ where d is the length of the longest chain in P_{s+1} strictly below s and u is the length of the longest chain strictly above s . To complete the proof we need only argue that $A_{d,u}^{s+1}$ is an

antichain. Suppose, for instance, $s < x$. Then there is a chain of $u + 1$ elements strictly above x . So $s \notin A_{d,u}^{s+1}$ after all.

- (iii) (Sketch.) We must build (P, R) with computable realizer (L_1, L_2) to diagonalise over all sets $A_1, \dots, A_n$ of computable antichains with $n = \binom{h+1}{2} - 1$, and this is done by induction on h . Let $h = g + 1$. For a particular $A_1, \dots, A_n$ we proceed as follows, again following Kierstead's account [120]. We concentrate on a single requirement.

At stage s , the construction is in two steps, with $P = P_1 \cap P_2$ where P_i are the sets of points added in step 1. In step 1 our goal is to arrange that the top h points T_s of P_1^s in L_1^s are all incomparable in R_s yet we force all the elements of T_s into different antichains.

To achieve this, suppose the points Q_r^s from P_1^s have been played so that $P \restriction Q_r^s$ has height $\leq r$, that the top r points T_r^s of Q_r^s in L_1 are all incomparable in R and we've forced exactly r antichains from $A_1^s \cdots A_n^s$ to cover Q_r^s , each of which has an element of T_r^s . Now we add new points $c_1, \dots, c_j$ into P_1^s one at a time so that c_i is the $(r + 1)$ -st point of L_1^s and the top point of L_2^s (currently) until the requirement is forced to put some c_j into A_k^s . This occurs by the point when c_j is in a chain of length $r + 1$ as $c_1 \cdots c_j$ is a chain. In this way via the substeps r we get to the first goal.

In the second step we ensure that $(P_2^s, R \restriction P_2^s)$ has height g , every element of P_2^s is under every element of T_s in R_s and incomparable in R_s to all elements of $P_1^s - T_s$; finally at least $\binom{g+1}{2}$ computable antichains are needed to cover P_2^s . It then follows that we can get $A_1 \cdots A_n$ since we'll need $h + \binom{g+1}{2}$ of them to cover and $h + \binom{g+1}{2} = \binom{h+1}{2}$.

To achieve this second goal, we simply use the inductive strategy we used for g , defining the new ordering in the obvious way: all new points x in P_2^s are over ones in $P_1^s - T^s$ and under all points in T_s in L_1^s and putting x under all points of P_1^s in L_2^s .

Again, there is no problem with using the layering technique here so the strategies combine as usual. $\square$

Corollary 6.20 (Szemerédi (1982), in Kierstead [120]) *For any m there is a computable ordered set of width m that cannot be covered by fewer than $\binom{m+1}{2}$ computable chains.*

Proof. We use Theorem 6.19 (iii). Let (P, R) be computable dimension 2 computable ordered set with computable antichain number $\binom{m+1}{2}$. Define P^* to be $(P, L_1 \cap L_2^*)$ where L_2^* is the dual of L_2 . A is a chain (antichain) in P^* iff A is an antichain (chain) in P . The width of P^* is m and P^* cannot be covered by $< \binom{m+1}{2}$ computable chains. $\square$

Using a different argument, Kierstead improved Corollary 6.20 for $m = 2$.

Theorem 6.21 (Kierstead [117, 120]) *There is a computable poset of width 2 which cannot be covered by fewer than 5 computable chains.*

Other results in this area complement our earlier negative ones as follows:

Theorem 6.22 (Kierstead, McNulty and Trotter [122]) *Every computable crown-free ordering which can be covered by at most c computable chains has computable dimension at most $c!$ (compare with Theorem 6.15).*

Theorem 6.23 (Kierstead and Trotter [123]) *Every computable interval order of width m can be covered by $3m - 2$ computable chains, and there are computable interval orders that cannot be covered by fewer than $3m - 2$ computable chains.*

Theorem 6.24 (Hopkins [100], see Kierstead [120]) *Every computable interval order of width m has computable dimension at most $4m - 4$ (compare with Theorem 6.16).*

The arguments above, particularly Theorems 6.22 and 6.23 are quite involved and we don't have space here to give a fair treatment of them. The reader should consult particularly Kierstead [120] for details. One last result we will prove is the following that says the width, computable chain cover number c and dimension are related if the computable dimension is bounded.

Theorem 6.25 (Kierstead, McNulty and Trotter [122]) *If (P, R) is a computable ordered set with $d(P) = d$ and P has width m , then $c(P) \leq \binom{m+1}{2}^{d-1}$.*

Proof. Use induction on d . Let $d = e + 1$ and $L_1, \dots, L_{e-1}$ a computable realizer of P . Let $Q = L_1 \cap \dots \cap L_e \cap L_{e+1}^*$ with L_{e+1}^* the dual of L_{e+1} . Then Q has height bounded by m . By Schmerl's result, A can be covered by $\binom{m+1}{2}$ computable antichains, so P can be (in Q). If A is a computable

antichain in Q then $P \upharpoonright A$ is a computable ordered set of width $\leq m$ and has realizer $(L_1 \upharpoonright A, \dots, L_e \upharpoonright A)$. Now we can use the induction hypothesis to cover each of the $\binom{m+2}{2}$ computable antichains covering P by $\binom{m+1}{2}^{e-1}$ computable chains. $\square$

We close with a few comments and a list of further questions. First Kierstead pointed out that there is a huge gap between Szemerédi's quadratic lower bounds and the exponential upper ones (for $c(P)$).

Questions 6.10

- (i) (Kierstead) *Let $b(m)$ be the maximum $c(P)$ over all computable P of width m . Is $b(m)$ polynomial? What is $b(2)$?*
- (ii) (Kierstead, McNulty and Trotter) *Is $d(P)$ finite if P has finite width and is 3-crownfree?*

Another possible approach is to examine dimension and covering numbers in terms of the arithmetical hierarchy. For instance, we would define the Π_n dimension (of A) to be the least m with $A = L_1 \cap \dots \cap L_m$ and the L_i all Π_n linear orderings. (Similarly we could do this for covering number.)

Question 6.11 *Investigate this notion. For example, is there a finite n with the Π_n dimension of an ordering the same as the classical dimension?*

Question 6.12 *How do the (degrees of) realizers relate to known notions in recursion theory such as Π_1^0 classes, etc., (e.g., Jockusch-Soare [104]) and to reverse mathematics? (See Simpson [222]; for related results see Hirst [98].)*

7 Degree-Theoretical Results, Jump Degrees and Worker Type Arguments

We return now to the problem of assigning a degree to the isomorphism type of a linear ordering. The natural approach is to define the degree of an order type to be the minimum of the degrees of the orderings of that type. By Richter's [201] result we met in Section 2 we know that if an ordering has a degree $\mathbf{a}$ then $\mathbf{a} = \mathbf{0}$. There are a number of possible extensions to this idea of degree which are useful for order types, and we shall look at some in this section. The first was suggested by Jockusch.

Definition 7.1 Let $\mathbf{A}$ be a structure. Define the n (-th jump) degree of $\mathbf{A}$ (for n a computable ordinal) to be $\min\{(\deg(\mathbf{B}))^{(n)} : \mathbf{B} \cong \mathbf{A}\}$, where $\mathbf{a}^{(n)}$ denotes the n -th jump of $\mathbf{a}$.

We shall say an ordering A has *proper n (-th jump) degree $\mathbf{a}$* if A does not have k -degree for any $k < n$ and A has n -degree $\mathbf{a}$. Julia Knight asked if a linear ordering could have proper 1-degree $\mathbf{0}'$. The reader should note that this is of some technical interest since the Feiner coding techniques of Section 2 (see Remark 2.1) construct Π_1 orderings not isomorphic to *low* orderings. Here the reader should recall that a set A (degree $\mathbf{a}$) is *low* if $A' \equiv_T \emptyset'$ (resp. $\mathbf{a}' = \mathbf{0}'$). This question was answered affirmatively by Jockusch and Soare, who introduced a new idea into these studies: the idea of separators with requirements played between them. This idea has some reflections in some of the earlier work of Watnick and Lerman but is rather different.

Theorem 7.1 (Jockusch and Soare [107])

- (i) *If B is computably enumerable and $B \not\equiv_T \emptyset$, then there exists a linear ordering A with $A \equiv_T B$ such that if $C \cong A$ then $C \not\equiv_T \emptyset$.*
- (ii) *In particular, if B is chosen low then it follows that there is an A of proper 1-degree $\mathbf{0}'$.*

Proof. We first sketch the proof without permitting (which keeps $A \leq_T B$). We build a Π_1 ordering $A (\subseteq \mathbb{Q})$ so that

$$R_e : A \not\equiv_T W_e.$$

First we concentrate on the basic module (for a single R_e). To meet R_e , we play a game with W_e . We first enumerate a point a_0 in A and wait till there occurs some point b_0 in W_e . Now we proceed as follows. We begin to build an ω^* sequence in A_e , $a_{n(s)}^s < \dots < a_2^s <^s a_1^s <^s a_0$ until W_e responds with a point $b_1 < b_0$. Note that if no such b_1 occurs in W_e then we win R_e as every point in A has infinitely many predecessors yet b_0 does not in W_e . Now if W_e responds at stage s_1 with $b_1 < b_0$ in W_{e,s_1} , we define *all of* $a_{n(s)}^s, \dots, s_1^s, a_0$ to be equal and define $a_1^{s_1} > a_0$ and begin a new ω^* sequence in A immediately above a_0 but below a_1^t as below in Figure 9.

$$a_0 <^t a_{n(t)}^t <^t \dots <^t a_1^t.$$

Figure 9.

Again we wait for there to occur some $b_2 < b_0$ in W_{e,s_2} for some $s_2 \geq s_1$. If no such b_2 occurs we win as A has order type $1 + \omega^*$ yet b_0 is not the last point of W_e but it has only one predecessor. The inductive strategies continue in the obvious way. That is, at any stage q we have

$$a_0 < a_1 < a_2 < \cdots < a_{k-1} < a_{n(q)}^q < \cdots < a_{k+1}^q <^q a_k.$$

Now if ever W_e does not respond then we build a $(k-1) + \omega^*$ sequence for A and win as b_0 has k predecessors. If W_e *always* responds then we win as we build an ω sequence *yet* b_0 *has infinitely many predecessors*.

The above concludes the basic module. Now to put the actions together for many R_e , Jockusch and Soare came up with the novel idea of a static set of *separators*. The final order type of A will be of the form

$$A_1 + S_1 + A_2 + S_2 + \cdots$$

where S_i is $1 + \eta + 1 + i + 1 + \eta + 1$. Now if $W_e \cong A$ then we must have S_i for all i . The idea is that we use the interval between S_{e-1} and S_e to meet R_e . Suppose $W_e \cong A$. Let (a, b, c, d) represent the separator S_i in W_e . That is, (a, b, c, d) are the points with a the left 1, (b, c) the $i+2$ -block and d the right 1. Then the relation

$$R(a, b, c, d) = \text{“}(a, b, c, d) \text{ represent } S_i \text{ in } W_e\text{”}$$

is a Π_2 relation uniformly in i , and so we can handle it on a Π_2 tree of strategies, as in Watnick's theorem of Section 6.

Thus for the e -module we attempt to do the basic module between S_{e-1} and S_e . Now in W_e we guess the quadruples q_1 and q_2 (or more neatly, the pair (q_1, q_2)) corresponding to S_{e-1} and S_e . Based on this belief we proceed as above, building A . The secret is that if ever a guess appears wrong (via a higher priority guess appearing correct) we can always cancel our mistakes: after all A is not computable so we can declare all the points based on weaker guesses to be equal. As usual, guesses that are too strong give only finitely many points wrong and by this observation the others don't matter so the tree q_1, q_2 can be used to win, if they exist.

We keep $A \leq_T B$ as follows. We only ask that if $B_s[k] = B[k]$ and $j < k$ then $a_j <^s a_k$ implies $a_j < a_k$ in A . That is, we can only change the ordering if the indices are permitted. This is an essentially routine modification. For instance, for a_0 we build our ω^* sequence at each stage s ,

$$a_{n(s)} <^s \cdots <^s a_1 <^s a_0$$

until we see some $m < n(s)$ with $b_m < b_{m-1} < \dots < b_e < b_0$ and $B_{s+1}[m] \neq B_s[m]$. Then we cut the a_0 -predecessors back to

$$a_{m-1} <^{s+1} \dots <^{s+1} a_0 <^{s+1} \dots <^{s+1} a_m.$$

This clearly causes no real hardships and we need only note that if points are placed for α and later for β and $\alpha <_L \beta$ then the points for β are bigger than those for α . Thus there is no real interference caused by the various e -modules between S_{e-1} and S_e . One point that should be noted is that the R_e -module must act at other than stages when it looks correct, in the “cutback” sense. We must only ensure that new points are defined only at stages when it looks correct.

The proof outlined above only gives $A \leq_T B$. To get $A \equiv_T B$, Jockusch and Soare proved the following:

Theorem 7.2 (Special case of Knight [124, Theorem 4.1]) *Let B be any set and suppose A is a linear ordering with $A \leq_T B$. Then there is a $C \cong A$ with $C \equiv_T B$.*

Proof (Jockusch and Soare [107]). We code B into C . To do this we ensure that, for all k , $c_{2k} <_C c_{2k+1}$ iff $k \in B$. Define an isomorphism f via: if $(k \in B \text{ iff } a_{2k} <_A a_{2k+1})$ let $f(a_{2k}) = c_{2k}$, $f(a_{2k+1}) = c_{2k+1}$; otherwise, let $f(a_{2k}) = c_{2k+1}$ and $f(a_{2k+1}) = c_{2k}$. $\square$

The reader might wonder if the hypothesis that B has computably enumerable degree is necessary in Theorem 7.1. If a set $B \leq_T \emptyset'$, we know that it has a computable approximation by the limit lemma. This leads to the technique of “ Δ_2 permitting”. We have already seen this in the proof that if A is a Δ_2 linear ordering then A is isomorphic to a Π_1 one. However, we recall what this means. We have $B = \lim_s B_s$ with $B_s(x) \neq B_{s+1}(x)$ only finitely often. We then construct $C \leq_T B$ by simple Δ_2 permitting by asking that if $B_s[x] = B[x]$ then $C_s[x] = C[x]$. Basically this means that if we think we have a permission we can do something. Then if later this permission is withdrawn, then we must “go back”. We must ensure that the construction works in the “true stages” of b ’s enumeration. The details are a little messy but, using this idea, the author, and, independently, D. Seetapun showed the following:

Theorem 7.3 (Downey, Seetapun) *If $\mathbf{a}$ is a Δ_2 nonzero degree then $\mathbf{a}$ contains a linear ordering isomorphic to a computable one.*

Julia Knight improved some earlier work of Ash, Knight and Jockusch to show that a consequence of Theorem 7.3 is

Theorem 7.4 (Knight) *If $\mathbf{a} \neq \mathbf{0}$ is any degree then $\mathbf{a}$ contains a linear ordering not isomorphic to a computable one.*

Proof. Let C be given. We show how to construct an ordering computable from C not isomorphic to a computable one. The proof splits into 3 cases.

CASE 1, $\emptyset'' <_T C''$: This case uses some work of Ash, Jockusch and Knight [9], (or of Lerman [138]). Define the $\sigma(X)$ to be the “shuffle sum” of X consisting of densely many copies of $\mathbb{N}$ and of n for $n \in X$. Then it turns out that $\sigma(C'' \oplus \overline{C''})$ can be chosen to have degree X (using a proof similar to Watnick’s Theorem), and cannot be isomorphic to a computable linear ordering since $\emptyset'' <_T C''$ (see Ash, Jockusch, Knight [9] for details.)

CASE 2, $C \leq_T \emptyset'$ and is noncomputable: This is the Downey-Seetapun Theorem.

CASE 3, $C'' \equiv_T \emptyset''$ and $\emptyset' <_T C'$: Using Case 2 in relativised form, we see that C' contains a linear ordering A not isomorphic to a computable one. Then $\gamma(A)$ can be chosen to have degree C and cannot be isomorphic to a computable ordering. $\square$

The reader should note that the Jockusch-Soare Theorem produces a Π_1 linear ordering. Richter’s result also shows that if A is any Δ_2 linear ordering with $A \not\equiv_T \emptyset$ then there is a Δ_2 linear ordering B with $B \cong A$ and $B \not\equiv_T A$. The only hope therefore of any degree for Π_1 linear orderings is perhaps to define the Π_1 degree of A to be:

$$\min\{\deg(B) : B \text{ is a } \Pi_1 \text{ linear ordering and } B \cong A\}.$$

Theorem 7.5 (Downey, see Downey and Moses [59]) *If $\mathbf{a}$ is any computably enumerable degree, then there is a Π_1 linear ordering with Π_1 degree $\mathbf{a}$.*

Proof. This was not actually written out in Downey and Moses [59], but we give a sketch here. This uses a variation on the technique of Theorem 7.1. We build $A = \cap_s A_s$ with:

$$R_e : \overline{W}_e \cong A \text{ implies } B \leq_T \overline{W}_e$$

where B is a given computably enumerable set. To make life easier, we replace B by $C = \cup_s C_s$ defined as follows. Let $f(\omega) = B$ be ε 1-1 computable function. At each stage s , let $c_{i,s}$ list $\overline{C}_s$ in order. Define

$$C_{s+1} = C_s \cup \{c_{f(s),s}, \dots, c_{f(s)+s,s}\}.$$

Then $C \equiv_T B$ and has nice properties (“retraceability” is the one we are interested in).

For a single R_e , we build a reduction $\Gamma(W_e) = A$ if we fail to ensure that $\overline{W}_e \not\equiv A$. We make $A \equiv_T B$ so this does the job. Again we begin by building an ω^* sequence below a_0 . Until $\overline{W}_{e,s}$ “responds appropriately” we then act in the *coding mode*. That is, we have $a_{n(s),s}, \dots, a_{j,s}, a_{j-1,s}, \dots, a_{0,s}$, and have $a_{j,s}$ the coding location of j . Now if j occurs in $C_s - C_{s-1}$ (and so all y in C_s with $s \geq y \geq j$), then we set $a_{j,s} = a_{j-1,s}$. (We assume $0 \notin C$.) Clearly if $\overline{W}_{e,s}$ does not act and this is the only action, then this section is $\equiv_T B$. Since some W_e are empty, this keeps $A \geq_T B$ via Γ as we will see.

Now as in Theorem 7.1 we need do nothing different until some j and k occur with $j < k$ and

$$b_k^s < \dots < b_0^s \in \overline{W}_{e,s} \quad \text{and} \quad j \in C_s - C_{s-1} \quad \text{with } j \text{ least.}$$

If (j, k) occur, then as in Theorem 7.1 we begin our new ω^* sequence to the right of a_0 . Obviously if this repeats without interruption, then we must win since $\overline{W}_e \not\equiv A$. The trouble is that in our case, $\overline{W}_e$ is not necessarily computable, *and it too can change*.

First we ask that whilst it appears we’ve achieved a (j, k) -action as above, say at stage s , and at stage $t > s$ we see some $i < j$ occur in $C_t - C_{t-1}$ then even if $b_k^t = b_k^s$, we will still cut the sequence down to $a_{i-1}^t \dots a_0$ and begin a_i^t to the right of a_0 as this action is more likely to succeed. (Note that for $q \geq j$ we are defining $a_q^{t-1} = a_0$ in this case.) See Figure 10.

$$a_{j-1}^{t-1} \dots a_0 \ a_{n(t)}^{t-1} \dots a_j^{t-1}$$

becomes

$$a_{i-1}^t \dots a_0 \ a_{n(t)+1}^t \dots a_i^t$$

Figure 10.

We can only be interrupted as follows. It may be that b_d^s enters $W_e - W_e^s$ for (many) $d \leq k$. A typical scenario is given in Figure 11.

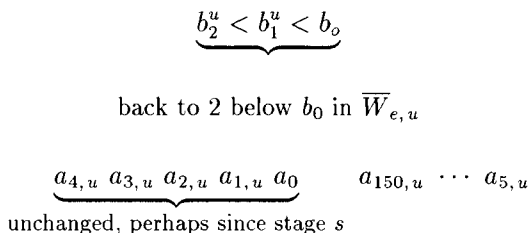


Figure 11.

The idea is that this gives progress on $C \leq_T W_e$. To implement this, we now go *back* to the original strategy holding $a_{150+m,u+1} < a_{4,u}$ for $m = 1, 2, \dots$ and enumerate an axiom for the reduction Γ , when $\overline{W}_e$ responds. That is, we define $\Gamma(b_{150+m}^{t_1}) = a_{m,t_1}$ for all $2 < m < 150$ and $\Gamma(b_j^{t_1} (= b_j^u)) = a_{j,t_1}$ for $j \leq 2$ at the least stage t_1 when $\overline{W}_{e,t_1}$ has ≥ 150 elements below b_0 . Of course if C permits below 2 before this, we don't need to worry, and if C permits below 150 before this we can put more of the a_i 's below a_4 . The point is that if there are infinitely many unsuccessful attacks trying to win below a_0 in this way, then Γ will eventually witness that $W_e \not\leq_T C$. The necessary details fit together as before. We should remark that this is a different procedure than that used in Downey-Moses [59]. $\square$

We remark that not all Π_1 linear orderings have Π_1 degrees.

Theorem 7.6 (Downey) *Given computably enumerable $C \not\leq_T \emptyset$, there is a Π_1 linear ordering $A \leq_T C$ such that A is not computable and A has no Π_1 degree. In fact, the Jockusch-Soare ordering of Theorem 7.1 has this property.*

Proof. We build $C_e = \lim_s C_{e,s}$ to meet:

$$R_e : W_e \equiv_T \emptyset \text{ or there is a } \Pi_1 C_e \text{ with } C_e \cong A \text{ and } W_e \not\leq_T C_e.$$

We split R_e into infinitely many $R_{e,i}$:

$$\begin{array}{l}
 R_e : W_e \equiv_T \emptyset \text{ or there is a } \Pi_1 C_e \text{ with } C_e \cong A \text{ and } (\forall i)(R_{e,i}) \text{ with} \\
 R_{e,i} : \Phi_i(C_e) \neq W_e.
 \end{array}$$

Let $1(e, i, s) = \max \{x : (\forall y < x)(\Phi_{i,s}(C_{e,s}; y) = W_{e,s}(y))\}$. We shall use Sacks preservation on $u(e, i, s) = \max \{e(\Phi_{i,s}(C_{e,s}; x) : x \leq 1(e, i, s))\}$. To do this we will need a Π_2 isomorphism to witness $C_e \cong A$. That is, we build a computable finitely branching tree of possible isomorphisms and the relevant isomorphism will be given by the leftmost path so that its graph is Π_2 .

Remember the Theorem 7.1 strategy is to build A_i 's between separators S_j of the form $1 + \eta + j + \eta + 1$. The idea is that we will allow $\langle e, i \rangle$ to move the isomorphism on some numbers x in the field of $C_{e,s}$ (i.e., $\text{dom } A_s$) with certain provisions. So $f_s : C_{e,s} \rightarrow A_s$. Now the idea is that if we have $f_s(z) \downarrow$ for some $z < u(e, i, s)$ we wish to keep $z \in C_e$. The trouble of course is that $f_s(z)$ may leave A_s , i.e., $f_s(z) \notin A_{s+1}$. We have only two options. Either we delete z from $C_{e,s+1}$ or we shift the isomorphism. The obvious place to shift such a $f_s(z)$ is into the η -part of some separator. Then we will never need to delete z from C_e . Unfortunately, if we do this infinitely often, for some fixed A_i , then C_e will not be isomorphic to A since the "1" from $1 + \eta + i + \eta + 1$ may be missing.

To overcome this problem, we will only allow $R_{e,i}$ to shift f_e on A_j for $j > \langle e, i \rangle$. So for A_j with $j \leq \langle e, i \rangle$ we will take the "delete z " option. This creates a few problems. Now we only need to build $C_e \cong A$ if $W_e \not\equiv_T \emptyset$. So the argument is that such $R_{e,i}$ only have finite effect if $W_e \equiv_T \emptyset$. First we don't have a problem arguing that $\Phi_i(C_e) \neq W_e$ with this: Since the only A_j that use the "delete z " option (and hence potentially change C_e on $u(e, i, s)$) are those with $j < \langle e, i \rangle$, then the Π_2 correct version of $R_{e,i}$ can know exactly which of the A_j for $j < \langle e, i \rangle$ act infinitely often. Since we know exactly how it will act, we can wait till $F_i(C_e; y) = W_{e,s}(y)$ via j -correct computations for $j < \langle e, i \rangle$. Since j -correct computation can be preserved (by shifting), this means that if $W_e \not\equiv \emptyset$, then $\Phi_i(C_e) \neq W_e$.

However, the problems are created by the "false" $R_{e,i}$ (those with incorrect guesses). We focus on $k \leq \langle e, i \rangle < j$. The problem comes in trying to keep $C_e \cong A$. For instance, we might have some Φ_i computation with some $z < u(e, i, s)$ for which we shift the isomorphism so that z is part of a S_j separator. We would add some new elements to pick up the A_j^s -computations (or A_{j-1} depending on which way we moved $f_s(z)$). The trouble is that some $z_1 < z$ in A_k^s , some $k < \langle e, i \rangle$, may be false. So later we remove z_1 from A_k^t . If this recurs infinitely often for the same Φ_i -computation (which we meet by divergence) we might fail to get an isomorphism on A_j . The idea is that when this occurs, we delete the new points we added to $C_{e,s}$ and put $f_s(z)$ back to where it originally was.

Using the true path method we can see that this designs a Π_2 isomorphism from C_e to A provided that we meet the $R_{e,i}$ on the true path. Suppose then that for some $R_{e,i}$ we have $\langle e, i \rangle$ -correct computations. This in the view that we build a tree of potential isomorphisms and if β is the true path then f_β is the correct f . Now if we move $f_s(z)$ for an $\langle e, i \rangle$ -correct computation, we won't need to move it back. If then the true version of $R_{e,i}$ acts infinitely often then we see, in the usual way, that W_e is computable (to compute $W_e(z)$ go to a stage where we have $1(e, i, s) > z$ via $\langle e, i \rangle$ -correct computations). $\square$

We remark that the above uses Π_2 isomorphisms which are different from all of our other proofs which essentially construct Δ_2 ones. It is not clear if we can modify the above proof to show

Question 7.1 *Does the Jockusch-Soare ordering of Theorem 7.1 (for any C) have the additional property that if B is a Δ_2 and isomorphic to A then there is a $\Pi_1 D \cong A$ with $B \not\leq_T D$?*

Another open question here is:

Question 7.2 *Is there a Π_1 linear ordering A with no computable copy but A has copies of each computably enumerable (Δ_2 ?) non-zero degree? (This seems unlikely.)*

The original use of the separator technique by Downey and Moses was to look at the degrees of successivities in a computable linear ordering A . Those authors were motivated by the following questions of Remmel:

Question 7.3 (Remmel) *If A is a computable boolean algebra, does there exists $B \cong A$ with B computable and the atoms of B of incomplete Turing degree?*

Downey [53] has shown the answer to Question 7.3 to be “yes”.

The corresponding question for linear orderings asks if the successivity relation can ever be “intrinsically complete”.

Theorem 7.7 (Downey and Moses [59]) *If A is computably enumerable and noncomputable, then there exists a computable linear ordering B with $S(B) \equiv_T A$, where $S(B)$ denotes the successivities of B , such that if D is a computable linear ordering with $D \cong A$ then $S(D) \not\equiv_T A$. In particular, if $A \equiv_T \emptyset'$ then $S(D) \equiv_T \emptyset'$.*

Proof. The original proof made direct use of separators. However, armed with the Downey-Knight result Theorem 2.9, we can give a proof using Theorem 7.5. Recall that this concerned $\gamma(Q)$ where for a linear ordering Q ,

$$\gamma(Q) = Q(\eta + 2 + \eta).$$

The result was that Q is Π_{n+1} iff $\gamma(Q)$ is Π_n . Apply this result via Theorem 7.5 to any linear ordering C of Π_1 degree $\deg(A)$. Then $B = \gamma(C)$ is computable and, by construction, $S(B) \equiv_T A$ and if $D \cong B$ is computable then $S(D) \equiv_T A$ by Theorem 7.5 and the iff in Theorem 2.9. $\square$

If we similarly apply Theorem 2.9 to Theorem 7.6 we see

Theorem 7.8 (Downey) *There is a recursive linear ordering A such that $S(A)$ does not have a degree. That is, $\min \{ \deg(S(B)) : B \text{ is a computable linear ordering with } B \cong A \}$ does not exist. Furthermore, A can be chosen with $S(A)$ of any non-zero computably enumerable degree.*

Question 7.4 *Develop a theory of “intrinsic degree” for other relations such as the block relation $B(A)$ on computable linear orderings. Develop a model theoretic setting as in Ash-Nerode [13].*

There have been some investigations into the interaction of $\deg(B(A))$ and $\deg(A)$. For instance: using rather more elaborate techniques, based on the separator idea, Jockusch and Soare showed:

Theorem 7.9 (Jockusch and Soare [107]) *There is a Π_1 linear ordering A such that $\langle A, B(A) \rangle$ is of low degree, yet A is not isomorphic to a computable linear ordering.*

Jockusch and Soare claimed in [107] that there is no common generalisation of Theorems 7.1 and 7.9. That is, *there is a computably enumerable degree $\mathbf{a} \neq \mathbf{0}$ such that if A is a linear ordering with $\deg \langle A, B(A) \rangle \leq \mathbf{a}$ then A is isomorphic to a computable linear ordering.* No proof of this result has yet appeared in print.

Question 7.5 *Is there a property P of classical order types that guarantees that if A has type τ , then A is low and $P(\tau)$ holds, then A is isomorphic to a computable linear ordering?*

One property is known:

Theorem 7.10 (Downey and Moses [59]) *Every low (indeed semilow) discrete linear ordering A has a computable copy.*

Proof. Choose the canonical choice set B for A . Namely, put z_0 into B if z_0 is the least element in A . Then put z_1 into B if z_1 is the least with z_1 in A but not in the same block as z_0 , etc..

If A is semilow (i.e., $\{e : W_e \cap A \neq \emptyset\} \leq_T \emptyset'$) we then define for each triple (x, b, y) an computably enumerable set $W_{g(x,b,y)}$ via

$$b \in B \leftrightarrow (\forall x < b)(x \notin A \text{ and } b \in A \text{ or } (\forall y)(W_{g(x,b,y)} \cap A \neq \emptyset))$$

where g is given by the s - m - n theorem. As A is semilow we know there is a computable f such that $W_{g(x,b,y)} \cap A \neq \emptyset$ iff $|W_{f(x,b,y)}| = \infty$. Thus $x \in B$ iff $(\forall x < b)(x \notin A \text{ and } b \in A \text{ or } (\forall y)(|W_{f(x,b,y)}| = \infty))$ so that B is Π_2 . By Watnick's theorem, there is a computable Q with $C_F(Q) = B$. But then $Q \cong A$. $\square$

Similarly we can ask for a property P of computable linear orderings to guarantee that if B is a computable linear ordering with property P then there is a computable $A \cong B$ with $S(A)$ incomplete.

Theorem 7.11 (Downey and Moses [59]) *If B is a computable discrete linear ordering, then there is a computable $A \cong B$ with $S(A)$ incomplete. Indeed, given any Δ_2 $C \not\equiv \emptyset$ we can arrange for $S(C) \not\equiv_T C$.*

Sketch Proof. Again this uses a modification of the Watnick construction. That is, we know $B \cong \zeta\tau$ for some Π_2 τ . We then construct $A \cong \zeta\tau$, with A computable, and ensure that we make $\Phi_e(S(A)) \neq C$ for any e . This uses a Π_2 version of the Sacks strategy of preserving agreements. It is rather similar to arguments we have seen earlier, so we omit the details. $\square$

We remark that by Watnick's and Feiner's results there are computable discrete A with $S(A)$ "intrinsically non-low". This last result is tight, since using any easy finite injury argument one can show:

Theorem 7.12 (Downey and Moses [59]) *If A is a computable discrete linear ordering, then there exists a computable B with $A \cong B$ and $S(B)$ semilow.*

It is not clear if there is always a B with $A \cong B$ and $S(B)$ low. Finally, one other direction here is to look at *strong reducibilities*. For instance:

Theorem 7.13 (Downey and Moses [59]) *If A is a computable linear ordering then there is a computable copy B of A with $S(B)$ finite or hyperimmune, and hence, by Downey and Jockusch [52], not w.t.t.-cuppable (so certainly not w.t.t.-complete).*

The proof of Theorem 7.11 is again an easy finite injury argument that we omit.

Returning to our original line of investigation we can ask: what are the possible n -degrees of linear orderings? Richter proved that if $n = 0$ then $\mathbf{0}$ is the only possibility. Using an interesting forcing argument with a number of applications, one of which is a similar result, Knight proved:

Theorem 7.14 (Knight [124]) *If A has 1-degree $\mathbf{a}$ then $\mathbf{a} = \mathbf{0}'$.*

The proof of Theorem 7.14 uses a forcing argument which would take us a little far afield and so we omit it. We remark that we know of no “direct” proof of Theorem 7.14.

In view of Richter’s and Knight’s results, a possible suggestion would be that for all n , if A has n -degree $\mathbf{a}$ then $\mathbf{a} = \mathbf{0}^{(n)}$. This is false. It is reasonable that it should be false since the coding techniques we have used need two jumps to decode. Ash, Downey, Jockusch and Knight proved the following definitive result:

Theorem 7.15

- (i) (Ash, Jockusch, Knight [9]) *If $n \geq 2$ is any computable ordinal and $\mathbf{a}$ is any degree with $\mathbf{a} > \mathbf{0}^{(n)}$ then there is a linear ordering A with proper n -degree $\mathbf{a}$.*
- (ii) (Downey and Knight [55]) *For any computable ordinal n there is a linear ordering A with proper n -degree $\mathbf{0}^{(n)}$.*

Some of the proofs in Theorem 7.15 use a new technique (“workers”) which we will describe in the bulk of the rest of the section.

Nothing really new is needed for finite n (although worker arguments were used in, for example, Ash, Jockusch and Knight [9]). Suppose n is finite. We

consider (ii) first. Again we use the Downey-Knight result Theorem 2.9. Indeed, this is what it was designed for. We already have $n = 1$ and $n = 2$. Use Theorem 7.6 (or in this case Theorem 7.1) in relativised form to get a linear ordering $D (= D(X))$ low over X , but not isomorphic to any X -presentable linear ordering, and such that if Q is any X' -presentable linear ordering isomorphic to D then there is an X' -presentable linear ordering $R \cong Q$ such that $Q \not\leq_T R$, if we use Theorem 7.6). Now apply this to $X = \emptyset'$ via Theorem 2.9. Let $P = \gamma(D) = D(\eta + 2 + \eta)$. We claim P has proper 2-jump. Certainly the 2-jump degree of P is $\mathbf{0}''$. Now suppose that P has proper 1-jump degree $\mathbf{e}$. Then $\mathbf{0}' < \mathbf{e} < \deg(D)$. Let $E \cong P$ with $E' \in \mathbf{e}$. Then $E = \gamma(F)$, some $F \cong D$. Choose $R \not\leq_T F$ with $R \cong F$. Then $\gamma(R)$ is isomorphic to P and has the wrong jump as $\gamma(R)' \leq_T R'$. (One can also use Knight's result to get this.) If P has 1-st jump degree, it would need to be $\mathbf{0}'$ and so P would need a low copy E . Then $E = \gamma(F)$, some F of degree $\mathbf{0}'$.

Now, for arbitrary finite $n > 2$, we proceed as follows. We have done $n = 1$. For $n \geq 2$ choose D low over $\mathbf{0}^{(n)}$ not isomorphic to a $\mathbf{0}^{(n)}$ -presentable ordering and having the additional property of Theorem 7.6. Now we know there is a copy of $\gamma(D) <_T \mathbf{0}^{(n)}$. Applying the γ operator via Theorem 2.9 n times we get $\gamma^n(D) <_T \mathbf{0}'$. Claim $\gamma^n(D)$ has proper $n + 1$ -degree $\mathbf{0}^{(n+1)}$. Clearly it has $n + 1$ -degree, say $\mathbf{a}$. Then $\mathbf{a} \leq \deg(D)$. Applying Theorem 7.6 we get some B with $B \cong A$ and $\deg(B) \not\leq \mathbf{a}$. Then $\gamma^n(B)$ does not have the right n -th jump.

To get (i) for $\mathbf{a} > \mathbf{0}^{(2)}$, we really need to code some invariant set into the ordering. For instance, we might attempt to do the following. Let $\mathbf{a} > \mathbf{0}^{(2)}$ choose Y with $Y'' \in \mathbf{a}$ and choose linear ordering A with the Theorem 7.6 property and low over Y' . We'd like then to argue that $\gamma(A)$ has proper 2-degree Y'' . There seems no clear way to show this without more control over (e.g.) A , although it may be true. The problem is arguing that Y'' must be coded into all isomorphic copies of $\gamma(A)$. To overcome this we work with new orderings where we have more control over their order type and more control over an encoded set (to keep the n -jump up). Much of the machinery can be supplied from earlier sections.

For finite n , to get (i), since we are dealing with degrees $\mathbf{a} > \mathbf{0}^{(n)}$, we need a little care as to exactly which set we code into the ordering to code $\mathbf{a}$. We will use Lerman's [138] result. Recall that this involved $\tau(S)$ where if $S = \{a_0 < a_1 < \dots\}$ then

$$\tau(S) = a_0 + \zeta + a_1 + \zeta + \dots$$

and from Lerman's proof, it follows that

$$Dg(\tau(S)) = \{\deg(D) : S \text{ is computably enumerable in } D''\}$$

where $Dg(X) = \{\deg(D) : D \cong X\}$ for a nontrivial linear ordering X . To get an ordering with arbitrary 2-degree let $X \leq_T \emptyset''$ then, via jump inversion, choose D with $D'' \equiv_T X$. Let $A = \tau(X \oplus \overline{X})$ be of degree D . It is easy to see that A has 2-degree $\deg(X)$ by Lerman's result, and cannot have 1-st or 2-nd jump degree by the Richter-Knight results.

We remark that using an infinite injury argument, Ash, Jockusch and Knight used a different ordering to code sets. This is of independent interest and we state their result (whose proof is similar to others we have seen) below.

Theorem 7.16 (Ash, Jockusch, Knight [9]) *Define $\sigma(R)$ to be the "shuffle sum" of R . That is, $\sigma(R)$ is composed of densely many copies of ω and $n+1$ for $n \in R$. Then for any R ,*

$$Dg(\sigma(R)) = \{\deg(D) : R \text{ is computably enumerable in } D''\}.$$

Clearly we could have used $\sigma(R)$ in place of $\tau(R)$ in the above reasoning. We also need four other lemmas whose proofs are quite technical and which we omit.

Lemma 7.17 (Macintyre [145]) *For any computable ordinal s and any A with $\emptyset^{(s)} \leq_T A$ there is an α -generic set R with $R \oplus \emptyset^{(\alpha)} \equiv_T R^{(\alpha)} \equiv_T A$.*

(For our purposes, it is not really necessary to know what an α -generic set is, we only need the properties stated in Theorems 7.18 and 7.19 below.)

Theorem 7.18 (Ash, Jockusch, Knight [9]) *If R is $(\alpha+1)$ -generic and $C = \{D : R \text{ is computably enumerable in } D^{(\alpha)}\}$ then for any $B \leq_T D^{(\alpha)}$, for all $D \in C$, we have $B \leq_T D^{(\alpha)}$. It follows that $\{D^{(\alpha)} : D \in C\}$ has no element of least degree.*

Theorem 7.19 (Ash, Jockusch, Knight [9]) *Let $C = \{D : R \leq_T D^{(\alpha+1)}\}$ for some R . If $B \leq_T D^{(\alpha)}$ for all $D \in C$ then $B \leq_T \emptyset^{(\alpha)}$. Thus if R is not computable in $\emptyset^{(\alpha+1)}$ then $\{D^{(\alpha)} : D \in C\}$ has no element of least degree.*

Finally, we will need one lemma when we deal specifically with limit ordinals.

Lemma 7.20 (Ash, Jockusch, Knight [9]) *If α is a computable limit ordinal, let $\{\alpha_n\}_{n \in \mathbb{N}}$ be a fundamental sequence of α . Let $R \subseteq \mathbb{N} \times \mathbb{N}$, with $R_n = \{k : (n, k) \in R\}$. Let*

$$C = \{D : R_n \leq_T D^{(\alpha_n)} \text{ uniformly in } n\}.$$

Suppose that $B \leq_T D^{(\beta)}$ for all $D \in C$. If $\alpha_k > \beta$ and $T_k = \bigoplus_{n < k} S_n$ then $B \leq_T T_k^{(\beta)}$. Hence, if $\beta < \alpha_k$ and S_k is not computable $T_k^{(\alpha_k)}$, then $\{D^{(\beta)} : D \in C\}$ has no element of least degree.

To get a linear ordering with proper 3-jump $\mathbf{a} > \mathbf{0}^3$, we can proceed as follows. Let R be 3-generic with $R^{(3)} \equiv_T \emptyset^{(3)} \oplus R$ where R^3 has degree $\mathbf{a}$. Now, via Theorem 7.18, if

$$C = \{D : R \text{ is computably enumerable in } D''\}$$

then $\{D'' : D \in C\}$ has no element of least degree. Let $A_3 = \tau(S)$. Then $R \leq_T D^{(3)}$ for all $D \in C$ and as $R^{(3)} \equiv_T \emptyset^{(3)} \oplus R$, $R^{(3)} \leq_T D^{(3)}$ for all $D \in C$. It follows that A_3 has proper 3-degree.

For finite n we note the uniformities of the proof of Theorem 2.9 give:

Lemma 7.21 *For any R and $n \leq \omega$,*

- (i) $Dg(\gamma^n(\tau(R))) = \{\deg(D) : R \text{ is computably enumerable in } D^{(n+2)}\}$
- (ii) $Dg(\gamma^n(\tau(R \oplus \overline{R}))) = \{\deg(D) : R \leq_T D^{(n+2)}\}$

We remark that in Ash, Jockusch and Knight [9], they noted a similar generalisation of Watnick's theorem (c.f. Roy and Watnik [212]). For instance in (i) we'd get

$$Dg(\zeta^n(\tau(R))) = \{\deg(D) : R \text{ is computably enumerable in } D^{(2n+2)}\}.$$

Using Lemma 7.21 and the facts on n -generic sets, we get (ii) for all finite n :

Lemma 7.22 *For each n with $2 \leq n < \omega$ and $\mathbf{d} > \mathbf{0}^n$ there is an ordering $A_n = A_n(\mathbf{d})$ with proper n -degrees $\mathbf{d}$.*

Proof. For $n = m + 2$, let R have degree $\mathbf{d}$ and let D_0 be n -generic with $D^{(n)} \equiv_T R$. Let $A_n = \gamma^n(\tau(R \oplus \overline{R}))$. Let $C = \{D : R \leq_T D^{(n)}\}$. By Lemma 7.21 (ii), $Dg(A_n) = \{\deg(D) : D \in C\}$, and R is the element of least degree in C . Finally A_n does not have $n - 1$ degree by Theorem 7.19. $\square$

When we proceed to infinite cases we run into a problem with the above. The point is now that encoding a set as above is no longer a classical invariant. We need a new idea. For the case of limit ordinals the idea is to piece together smaller pieces in some fashion, and it is here we use Lemma 7.20. For instance, let $\alpha = \omega$, let $R \subseteq \mathbb{N} \times \mathbb{N}$ and $R_n = \{k : \langle n, k \rangle \in R\}$. Now let $\lambda(R) = \sum_{n < \omega} \gamma^n \tau(R_n)$. It is not difficult to see that the uniformities of the previous work show:

Lemma 7.23 $Dg(\lambda(R)) = \{\deg(D) : R_n \text{ is computably enumerable in } D^{(n+2)} \text{ uniformly in } n\}$.

We remark that Ash et al [9] used $\rho(R) = \sum_{n < \omega} \zeta^n \sigma(R_n)$ where:

Lemma 7.24 (Ash, Jockusch, Knight [9]) $Dg(\rho(R)) = \{\deg(D) : R_n \text{ is computably enumerable in } D^{(2n+2)} \text{ uniformly in } n\}$.

It is not hard to apply either Lemma 7.24 or 7.23 to any $\mathbf{d} > \mathbf{0}^{(\omega)}$ via Macintyre's result and Lemma 7.20. That is,

Theorem 7.25 (Ash, Jockusch, Knight [9]) *If $\mathbf{d} > \mathbf{0}^{(\omega)}$ there is an ordering A_ω with proper ω -degree $\mathbf{d}$.*

Proof. Choose ω -generic R with $R^{(\omega)} \equiv_T \emptyset^{(\omega)} \oplus R$ and $R^{(\omega)}$ of degree $\mathbf{d}$. We know for all m, k that $R_{k+1} \leq_T (R_0 \oplus \cdots \oplus R_k)^{(m)}$. Let $A_\omega = \lambda(R)$. If

$$C = \{D : R_k \text{ is computably enumerable in } D^{(2+k)} \text{ uniformly in } k\}$$

then by Lemma 7.20, $Dg(A_\omega) = \{\deg(D) : D \in C\}$. So A_ω has ω -jump $\mathbf{d}$. To see that it is proper, suppose it had n -degree for some n . Then there is some $D_0 \in C$ with $D_0^{(n)} \leq_T D^{(n)}$ for all $D \in C$. It follows, by Lemma 7.20, that if $k + 2 > n$, then

$$D_0^{(n)} \leq_T (R_0 \oplus \cdots \oplus R_{k-1})^{k+2}$$

uniformly in k . Hence $D_0^{(k+2)} \leq_T (R_0 \oplus \cdots \oplus R_{k-1})^{(m)}$, some m , and hence

$R_k \leq_T D_0^{(k+3)}$, so that

$$R_k \leq_T (R_0 \oplus \cdots \oplus R_{k-1})^{(m+1)},$$

a contradiction. $\square$

For $\alpha > \omega$ we really need something new. This is where the “worker” idea comes in. We will look in detail at the case $\alpha = \omega + 1$ (which gives the main idea) and only give some remarks on how one treats the general case.

First we wish to give the reader some intuition as to the “methods of workers” first introduced by Harrington [91] (see also Calhoun [27, 28] for a more formal presentation of these ideas). Consider the typical way we describe a priority argument. We tend to isolate the particular requirements and then discuss how the strategies cohere. This usually involves discussion of the true path (where we really would like to work) but then we must also discuss how the *computable version* of the construction works, where we only have approximations to the true path.

For instance, consider a typical injury priority argument. We call this more precisely a $\emptyset'$ -priority argument since we know that $\emptyset'$ can sort out exactly how we meet the requirements. Consider the result, Theorem 2.9, that R is X' -presentable iff $\gamma(R)$ is X -presentable. Let $X = \emptyset$ for the unrelativised version. Let $R \subseteq \mathbb{Q}$. For each $x \in R$ we wait to replace it with a successivity and put η 's between the successivity. If R were computable, this would pose no problem. In the worker model we have two workers, worker 1 and worker 0. Worker 1 works at the $\emptyset'$ level. It knows everything about R . Its task is to enumerate R as a subset of $\mathbb{Q}$ and tell worker 0 to make the $x \in R$ into successivities and put η 's between them. But, as often happens between management and the floor staff, worker 0 has difficulty understanding worker 1's commands. Worker 0 only lies at the computable level. By the limit lemma, worker 0 can only comprehend worker 1's command via a series of Δ_2 approximations. That is worker 0 has at its disposal a computable $f(x, s)$ where for all x , $\lim_s f(x, s) = R(x)$. So for x , worker 1 says either $x \in R$ or $x \notin R$. Worker 1 only gets $f(x, 0), f(x, 1), f(x, 2), \dots$. However, it knows that eventually we reach a t such that $f(x, t')$ is constant for all $t' > t$. The traditional “coherence” strategy of a priority argument can be paraphrased by saying that worker 0 can live with a finite number of mistakes. Of course we should always really represent $\overrightarrow{f(x, s)}$, i.e., $f(0, s), \dots, f(x, s)$, in place of $f(x, s)$ so we know the values *together*. This is important in the later more complex arguments.

Let us now consider a typical $\emptyset''$ argument. Since we have already a proof of it, let us again pick on Watnick's theorem. So now R is $\emptyset''$ -presented and we want ζR to be computably presentable. For such an argument we have 3 workers. Worker 2 works at the $\emptyset''$ level, 1 at the $\emptyset'$ level and 0 at the computable level. Worker 2 knows all of R , enumerates its points, issues commands such as "build ζ around each point".

Worker 1 can see worker 2's commands via the limit lemma. This worker uses $f(x, s)$ with $\lim_s f(x, s) = R(x)$ existing, f computable in $\emptyset'$ and when $f(x, s) = 1$ it says to worker 0 put ζ around x . When $f(x, s) = 0$ it says to worker 0, don't put ζ around x , and fix up your mistakes if $f(x, s-1) = 1$.

At the bottom level worker 0 receives 1's commands via the limit lemma, but now things are more complex. We know that f behaves such as, perhaps

$$f(x, 0) = 0, \quad f(x, 1) = 1, \quad f(x, 2) = 0, \quad f(x, 3) = 0, \quad \dots$$

Now g , which approximates 1's commands, needs to approximate the *binary* function f . So g can be wrong finitely often on each (x, i) . So typically $f(x, i) = j$ corresponds to a finite number of $g(x, i, \cdot)$ values. For instance, $g(x, i, 0), g(x, i, 1), \dots, g(x, i, s) = j$ and then $g(x, i, s) = j$ thereafter. This is why worker 0 only sees the correct value of $R(x)$ infinitely often.

Let us now see in detail how this actually works. One sleight-of-hand in the above presentation is that, at any stage s , we only deal with finite portions. So it is not correct to say worker 1 issues a command to build a ζ -piece around x . This is only "globally" true if the command is "really" true. (Note the use of the recursion theorem.)

At stage s , worker 2 computes $R_s \subseteq \{q_0, \dots, q_s\}$ with q_i the i -th member of $\mathbb{Q}$. We will replace the explicit use of the limit lemma by the expression "guesses". That is if $f(x, s) = 1$, as above, then worker 1 guesses $x \in R$.

Worker 1 guesses R_s . Let k_s be the first k with $k = k_{s-1} + 1$ or k the first place with the guess changing from $s-1$. The worker issues an instruction i_s via k_s . This either says destroy some previous work, or begin to build a ζ interval around k_s . The latter occurs if k_s now appears to be in R and the former if k_s was once in R but now is guessed out of R (i.e., currently appears to be out of R). These must cohere in the sense that we can carry out i_s in the environment left by $i_0, \dots, i_{s-1}$. That is, *we need to be able to correct mistakes*. By our analysis of Watnick's theorem in Section 4, we know how to destroy old work coherently: we incorporate it into other blocks. In the end, worker 1 produces an eventually stable set of commands $\{i_{j_0}, i_{j_1}, \dots\}$ which are not subsequently revoked and replaced with contradictory commands.

Worker 0 just works one level down. The way this is carried out is discussed in detail in Section 4 so we omit translation into this language. The crucial thing to keep in mind is that everybody needs to be able to be sufficiently conservative so that its *stable* set of actions (instructions) does the correct thing and the false commands or actions can be repaired.

So what is the point of this? First it gives another viewpoint as to how priority arguments work and a flexible model for constructions. At this stage, we should mention that there are several other formalisations of the idea of a “depth- α ” priority argument. One can use iterated trees of strategy (Slaman, and especially Lempp-Lerman [137]). Another approach via iterated trees is formalised in the work of Ash on α -systems (Ash [4, 6, 7, 8]). In any construction, of course, all we “really do” is approximate a $\mathbf{0}^{(\alpha)}$ predicate computably. (In some sense this idea is used in Feiner [67].) But this is an inadequate description since the crucial thing is to say how to organise the strategies to be able to get the requirements to cohere. This is why, for instance, the tree approach is often nicer than others since we put the “atomic outcomes” on the tree and can “see” what is happening. However, in many inductive instances, it is undesirable perhaps to use all the tree machinery and sometimes the worker approach seems the most perspicuous. Finally there have been a number of attempts to prove metatheorems for these depth n arguments. In particular, first Ash [4] (with α -systems) and later Knight [128, 127] developed rather general conditions under which one can construct a structure B obeying the appropriate list of conditions. The problem with these (in the opinion of the author) is that it often seems more difficult to verify the conditions than to perform a direct construction. The final verdict will be determined by what is the more useful.

Using arguments with infinitely many workers, Ash, Jockusch and Knight proved Theorem 7.15 (ii) as follows. Let α be a computable ordinal. Let $\nu^\alpha(R)$ be the sum of densely many copies of $q + 1 + \sum_{1 \leq \alpha < \beta} \zeta^\beta$ for $q \in \omega$, $i \leq \beta < \alpha$ and $r + 1 + \sum_{1 \leq \gamma < \alpha} \zeta^\beta$ for $r \in R$. Given a fundamental sequence $(\alpha_n)_{n \in \omega}$ for α , define a new sequence $(\alpha_n^*)_{n \in \mathbb{N}}$ via:

- (i) if $|\alpha_n| < \infty$, let α_n^* be the first $k \geq 3$ with $k \geq \alpha_n$ and if $n > 0$ then $k > \alpha_{n-1}$,
- (ii) if $|\alpha_n| = \infty$ and β is the greatest limit ordinal with $\beta \leq \alpha_n$, $\mu = \beta + k$ for some odd integer k , and if $n > 0$ then $\mu > \alpha_{n-1}$.

Definition 7.2 (Ash, Jockusch, Knight [9]) Let α be a computable ordinal and $R \subseteq \omega$. Define $A_\alpha(R)$ via

- (i) if $\alpha = 2m + 2 < \infty$ then $A_\alpha(R) = \zeta^m \sigma(R \oplus \overline{R})$.
- (ii) If $\alpha = 2m + 3 < \infty$ then $A_\alpha(R) = \zeta^m \sigma(R)$.
- (iii) If α is a limit ordinal and (α_n^*) is defined as above, then $A_\alpha(R) = \sum_{n \in \omega} (1 + \eta + 1 + A_{\alpha_n^*}(R_n =_{\text{def}} R \cap \omega^{(n)}))$.
- (iv) If β is a limit ordinal and $\alpha = \beta + 1$ then $A_\alpha(R) = \nu^c(R \oplus \overline{R})$.
- (v) If β is a limit ordinal and $\alpha = \beta + 2$ then $A_\alpha(R) = \nu^c(R)$.
- (vi) If β is a limit ordinal and $\alpha = \beta + 1 + 2k + 2$ then let $A_\alpha(R) = \zeta^{\beta+k} \sigma(R \oplus \overline{R})$.
- (vii) If β is a limit ordinal and $\alpha = \beta + 2k + 4$ then let $A_\alpha(R) = \zeta^{\beta+k} \sigma(R)$.

Lemma 7.26 (Ash, Jockusch, Knight [9]) Let $A_\alpha = A_\alpha(R)$

- (i) For finite even $\alpha = 2m + 2$,
 $Dg(A_\alpha) = \{\deg(D) : R \leq_T D^{(\alpha)}\}$.
- (ii) For finite odd $\alpha = 2m + 3$,
 $Dg(A_\alpha) = \{\deg(D) : R \text{ is computably enumerable in } D^{(\alpha)}\}$.
- (iii) For α limit,
 $Dg(A_\alpha) = \{\deg(D) : R_n \leq_T D_n(\alpha_n^*) \text{ uniformly in } n\}$.
- (iv) For $\alpha = \beta + 2k + 1$ with β limit and $k \in \omega$,
 $Dg(A_\alpha) = \{\deg(D) : R \leq_T D^{(\alpha)}\}$.
- (v) For $\alpha = \beta + 2k + 2$ with β limit and $k \in \omega$,
 $Dg(A_\alpha) = \{\deg(D) : R \text{ is computably enumerable in } D^{(\alpha)}\}$.

Applying Lemma 7.26 to an α -generic R via Theorems 7.18, 7.19 and Lemma 7.20, we get (in the same way as for finite n) Theorem 6.16 (i). We will not give the detailed proofs associated with Lemma 7.26. Rather we will briefly outline an ω -worker proof of a result from the Ash et al [9] paper:

If $D \leq_T \mathbf{0}^{(\omega+1)}$, then $\zeta^{(\omega)} D$ is computably presentable.

Sketch Proof. The proof of this infinite generalisation of Watnick's theorem uses one worker for each $n < \omega$ and two limit workers: ω and $\omega + 1$. Define the relation $\sim_n$ as follows:

$x \sim_0 y$ if there are only finitely many elements between x and y ,

$s \sim_{\alpha+1} y$ if there are only finitely many α -equivalence classes between $[x]_\alpha$ and $[y]_\alpha$,

and for limit β , $x \sim_\beta y$ if for some $\alpha < \beta$, $x \sim_\alpha y$.

Now worker $\omega + 1$ enumerates the ordering D . Worker ω is given an approximation to D . At stage s , this is those e such that for some $C \leq s$, C is a halting computation of $\Phi_e^{\emptyset^{(\omega)}}(e)$, in which there are no questions asked of $\emptyset^{(n)}$ for $n > s$. Let $\emptyset_s$ be this approximation. We see that $e \in \emptyset^{(\omega+1)}$ iff $e \in \emptyset_s$ for all sufficiently large s . The point is that worker ω can try to use $\emptyset_s$ instead of $\emptyset^{(\omega+1)}$ (to compute D via $\Gamma(\emptyset^{(\omega+1)}) = D$) and bounding computations by s . Then if we let $f^*(n, s)$ be the guess to $D(n)$ obtained this way, since $f^*(k, n)$ is uniformly computable in $\mathbf{0}^{(n)}$, we can let worker n operate at the $\mathbf{0}^{(n)}$ level and then worker n can compute $f^*(k, n)$ in the same way that worker ω can.

Now the strategy is rather straightforward; the Watnick strategy can be fully approximated so that worker $n + 2$ can issue commands to workers $n + 1$ and n in exactly the same way, but using the approximation given as above. Then worker $n + 2$ will succeed in commanding worker 0 to build ζ^n blocks. Finally the only difference is that when we correct at worker n then we correct via $\sim_n$ in place of $\sim_1$ and $\sim_0$ in Watnick's theorem. One then argues by induction, that for all n , worker n 's commands are carried out. $\square$

The actual details of these arguments seem very messy, and certainly in many applications the details are given in a rather informal way (but see Calhoun [27, 28]). The work of Ash and of Lempp-Lerman gives formal versions of arguments along similar lines.

For instance, Ash viewed the arguments as examples of metatheorems. Here is one example.

Let U and L be computably enumerable sets and P a computably enumerable tree consisting of finite nontrivial sequences $\ell_0 u_1 \ell_1 u_2 \dots$, with ℓ_0 and $\ell_n, u_{n+1} \in L \times U$. Let E be a computable partial function called the

enumeration function, with $E : L \mapsto \{F \subset \mathbb{N} : F \text{ finite}\}$. We let q be a Δ_α^0 function from the members of P of odd length (i.e., ending in L) such that if $q(\sigma) = u$ then $\sigma u \in P$. The function q is called the instruction function and corresponds to the α level worker in the worker argument. A path $\pi = \ell_0 u_1 \ell_1 u_2 \dots$ of P is called a *run* if it obeys the instruction function. That is, for all n , $q(\ell_0 \dots \ell_n) = n_{n+1}$.

Ash then defined an α -system $\mathcal{A}$ to be a structure

$$(L, U, \ell_0, P, E, \leq_\beta : \beta \leq \alpha)$$

as above for which additionally,

- (1) for all $\beta \leq_\beta$ is transitive and reflexive,
- (2) if $\gamma < \beta$ then $\ell \leq_\beta \hat{\ell}$ implies $\ell \leq_\gamma \hat{\ell}$,
- (3) if $\ell \leq_0 \hat{\ell}$ then $E(\ell) \subseteq E(\hat{\ell})$,
- (4) if $\sigma u \in P$, σ ends in ℓ^0 , $\ell^0 \leq \beta_0 \leq \ell^1 \leq_{\beta_1} \dots \leq \beta_{k-1} \ell^k$, and $\alpha > \beta_0 > \dots > \beta_k$, then there is an m such that $\sigma u m \in P$ and $\ell^i \leq_{\beta_i} m$.

Theorem 7.27 (Ash [5, 7, 8]) *Given an α -system $\mathcal{A}$, and q , a Δ_α^0 instruction function for P , there is a Δ_α^0 run π of (P, q) such that $E(\pi)$ is computably enumerable.*

The proof of the above works by induction for finite α and then along some system of notations for arbitrary computable α . For an application yet again we look at Watnick's Theorem. So we suppose that $(A \leq_A)$ is a ordering of type τ with $\tau \Delta_\alpha^0$ computable. Let C be a computable set upon which we will construct the computable linear ordering. For suitable ordering $(X, \leq_X)$ of type β (e.g., we have $\alpha = 2n$ and $\beta = \zeta^n$, as usual), we define T to be the tree of all nonempty finite sequences $(B_0, f_0, B_1, f_1, \dots)$ such that

- (i) $B_i = (B_i \leq_i)$ is a linear ordering, with $B_i = \{a_1, \dots, a_{k_i}\}$ for some k_i and $k_0 = 0$,
- (ii) B_{i+1} extends B_i ,

- (iii) $f_i : C \mapsto X \times B_i$ is a partial injective function,
- (iv) f_{i+1} extends f_i ,
- (v) $\text{dom } f_{i+1}$ contains the first i elements of C , and
- (vi) $\text{ran } f_{i+1}$ contains the first i elements of $X \times A$ which are in $X \times B_i$.

To use Theorem 7.27, we arrange that the ordering on C induced by $f = \cup_n f_n$ is computable, or equivalently computably enumerable. We define $E(B_0 f_0, \dots, B_n f_n)$ to be the finite ordering induced on C by f_n ; $\{(p, q) : p, q \in \text{dom } f_n \wedge f_n(p) \leq f_n(q) \in X \cdot A\}$. To then apply the meta-theorem we need suitable relations $\leq_\beta$.

This is achieved via so-called β maps, where we say that $g : \mathcal{A} \mapsto \mathcal{B}$ is a β map if, for $\text{dom } g = \{a_1, \dots, a_n\}$, then every Π_β sentence true $(\mathcal{A}, a_1, \dots, a_n)$ is also true in $\mathcal{B}$.

Then for

$$\sigma = (B_0, f_0, \dots, B_m, f_m)$$

and

$$\tau = (B_0, f'_0, \dots, B_n, f'_n),$$

define $\sigma \leq_\beta \tau$ iff $\text{dom } f_m \subseteq \text{dom } f'_n$, and $f'_n f_m^{-1}$ is a β map from $X \cdot B_m$ to $X \cdot B_n$.

The above is a very brief sketch of Ash's ideas and we urge the reader to look at Ash [8] and the forthcoming monograph Ash-Knight [11] for details. One of the problems with this and the even more elaborate Lempert-Lerman approach is that one of the must vary the metatheorem to get the details to work in new settings, and even then work must be done do actually verify the conditions.

We mention one recent application where such a variation was needed. Knight [129] who proved, that the set of limit points of a ordering of type ω^2 can have arbitrary 2-REA degree.

In any case, we have dealt with degrees strictly above $\mathbf{0}^{(\alpha)}$. To finish the proof, we need to get orderings of proper degree $\mathbf{0}^{(\alpha)}$ for $\alpha \geq \omega$. These are obtained via similar arguments where one comes up with an order type to code the appropriate sufficiently generic set (in the correct way). These results are rather technical and along similar lines, so we omit them (see Downey-Knight [55] for details). $\square$

Before leaving this topic, we mention one last result.

Theorem 7.28 (Ash, Jockusch, Knight [9]) *There is an ordering that does not have α -degree for any computable α .*

Sketch Proof. In Ash et al [9], the authors claim that one can show, via the forcing introduced in Ash, Knight, Manasse and Slaman [12], that ω_1^{ck} has no α -degree. $\square$

Question 7.6 *Under what conditions does an order type have an α -degree?*

One can use worker arguments for many other problems. One of the first applications to orderings as due to Ash [4, 6]. Here, Ash investigated conditions under which a computable model A was Δ_α stable. This means that for all computable $B \cong A$, $B \cong_\alpha A$. That is, all isomorphisms between B and A are Δ_α . This is not to be confused with Δ_α -categorical. This means that for any computable $B \cong A$ there is a Δ_α isomorphism between B on A . For instance $\mathbb{Q}$ is computably categorical yet is not Δ_α stable for any α (as there are uncountably many automorphisms of $\mathbb{Q}$). Now by our earlier results, we know that ω is not computably categorical, (and so not computably stable) yet it is easy to see that ω is Δ_2 stable. Recall that a computable structure A is *rigid* if it has no nontrivial automorphisms (Ash [4] observed that A is Δ_n stable iff for some set of constants $\{P_1, \dots, P_n\} = P$, (A, P) is rigid and Δ_n -categorical). Ash used his α -system arguments to characterise Δ_α -stable structures in terms of certain computable infinitary formulae.

Barker [15] did the same for Δ_n -categoricity. These results were subject to certain decidability conditions, which Ash, Knight, Manasse and Slaman [12] observed could be eliminated if the results were asked to hold absolutely for all relativisations. These results can be applied to linear orderings to show, for instance, ω^α is $\Delta_{\alpha+1}$ -stable for all α for all computable α (Ash [4, 6, 7, 8]).

As our final example of this technique we briefly outline a solution to a question of Rosenstein [208, 209] (also Lerman-Rosenstein [140]), using a worker argument.

Theorem 7.29 (Downey) *There is a computable, nonscattered linear ordering with no arithmetical dense subset.*

Sketch Proof. We build A of order type $\zeta^\omega\eta$. We must only ensure that if B is an infinite arithmetical subset of A then for some $x, y \in B$, $x \sim_n y$. We start with $q_0, q_1, \dots$, a copy of $\mathbb{Q}$. We must define an isomorphism f with domain $\mathbb{Q}$ (i.e., ensure that $f(q_i)$ exists for all i) and have a level $2n$ task of building ζ around each $f(q_i)$. The only requirement we need add is that if A_i is the i -th arithmetical set and $|A_i| = \infty$ then for some n , $x \sim_n y$ for some $x, y \in A$. We can enumerate the arithmetical sets as $A_0, A_1, \dots$ so that A_i is computable in $\mathbf{0}^{(i)}$. We let worker $2n+2$ diagonalise argument A_n . We do not let worker $2n+2$ change $f(q_i)$ for $i \leq 2n+2$. We let worker $2n+2$ act if it sees $x, y \in A_n$ with x, y not in the same block as $f(q_i)$ for $i \leq 2n+2$, and such that if we declare $x \sim_{2n+2} y$ then we won't amalgamate $f(q_i)$ and $f(q_j)$ for $i, j \leq 2n+2$. Worker $2n+2$ then asks that $x \sim_{2n+2} y$. We must then build a new copy of some $f(q_j)$ for some $j \geq 2n+2$. For instance if $x \sim f(q_j)$ and $y \sim f(q_k)$ then we might need to build a new $f(q_k)$ (e.g., if $k > j$). We can fix up errors by asking that if we begin a new $f(q_k)$ and later (e.g.) $y \in A_n$, then we cancel this (forever) and go back to the old one. The details go through in the standard way. $\square$

We close this treatment with a question suggested by Theorem 7.29.

Question 7.7 *Is there a computable non-scattered linear ordering not isomorphic to one with arithmetical dense subset?*

8 Other Work and Related Results

In this section we will look briefly at some other related work. Many of our arguments are diagonalisations. Jockusch and Soare [107] asked if there is an analogue of the recursion theorem for orderings. Let

$$L = \{e : \varphi_e \text{ is the characteristic function of a linear of } \omega\}$$

and let A_e denote $\langle \omega, \varphi_e \rangle$ if $e \in L$. Jockusch and Soare showed:

Theorem 8.1 (Jockusch and Soare [107]) *There is a computable f such that for all $e \in L$, $f(e) \in L$ and $A_{f(e)} \not\cong A_e$.*

They point out that the following analogue of the recursion theorem is, however, open.

Questions 8.1 (Jockusch and Soare [107]) *Is there a computable f with the following properties?*

- (i) $(\forall e)[e \in L \rightarrow f(e) \in L]$
- (ii) $(\forall e \in L)(A_{f(e)} \not\cong A_e)$
- (iii) f is well defined on order types. That is,
 $(\forall a, b)[(a, b \in L \text{ and } A_a \cong A_b) \rightarrow A_{f(a)} \cong A_{f(b)}].$

There is a lot of interesting related work on Boolean algebras. Remmel [197] is an excellent source here. Many analogous questions seem much harder for Boolean algebras since so many order types correspond to the same Boolean algebra, but things can differ. Downey and Jockusch [53] proved that every low boolean is isomorphic to a computable one. John Thurber demonstrated that each low₂ boolean algebra is also isomorphic to a computable one.

Question 8.2 *Develop results akin to those of Ash, Downey, Jockusch and Knight for boolean algebras. In particular, is there a low₃ boolean algebra not isomorphic to a computable one? Also what is the precise relationship between orderings and associated algebras? Finally, what is the effective content of the Ketonen invariants (Ketonen [110]) classifying countable boolean algebras?*

There is some work here. Jockusch and Soare have proven that no boolean algebra can have proper finite n -th jump degree $\mathbf{a} > \mathbf{0}^{(n)}$. But any degree $\mathbf{a}$ above $\mathbf{0}^\omega$ can be the proper ω -th jump degree of a boolean algebra. We refer the reader to Downey [51] for more on this topic.

Feiner [66, 67] showed that there were Π_1 boolean algebras not isomorphic to computable ones. Thus the argument will be much more complicated than that used for orderings. Feiner's result on Π_1 boolean algebras in some ways is a forerunner to the worker arguments we have seen in Section 7 and was a very fine achievement for the 1960's.

There are of course many other ordered structures we have not examined. For instance, Alex Feldman [70] has some interesting results on recursion theoretical aspects of a lower semilattice (rather along the lines of the Metakides-Nerode-Remmel approach of looking at lattices of substructures).

Another area we have not looked at is ordered structures, Metakides and Nerode [160] analysed Craven's [35] classification of the cone of orderings of

a formally real field. They showed that for any Π_1^0 class P , there is a computable formally real field whose orderings are in one-one correspondence with the members of P (in particular up to Turing degree). Thus, for instance,

Theorem 8.2 (Metakides and Nerode [160]) *There are computable formally real fields with no computable orderings and even some with all their orderings of incomparable Turing degrees.*

Question 8.3 *If A is a formally real computable field, is A isomorphic to a computably ordered computable formally real field?*

Downey and Kurtz [56] looked at the analogous result for ordered abelian groups. The question is nontrivial since it is not clear when the fact that a field A can be ordered as an additive group implies that it can be ordered as a field.

Theorem 8.3 (Downey and Kurtz [56]) *There is a torsion free computable abelian group G with no computable orderings. Indeed, G has infinite rank yet its only computably orderable subgroups have finite rank.*

Questions 8.4 (Downey and Kurtz [56])

- (i) *Can one construct a computable G whose orderings have exactly the degrees of a given Π_1^0 class?*
- (ii) *Is every orderable computable (abelian) group isomorphic to a computably orderable computable group?*

We remark that little is known about orderings of computable nonabelian groups. For instance, by a classical result of B. Neumann [176] any free group is orderable. Also by N. Neumann [177], every ordered group is isomorphic to a quotient of (an ordering of) a free group by a convex normal subgroup. Despite the highly noneffective classical proof, recently Reed Solomon has proven that there is a computable version of this result. However, many questions suggest themselves.

Question 8.5 *What is the effective content of Hölder's theorem [99] on Archimedean ordered groups?*

We refer to Solomon [234] for more details and for a number of very attractive results on computable ordered groups.

Another area that has attracted a great deal of attention recently is *subcomputable* or *feasible* algebra. Here one studies not computable, but polynomial time, analogues of classical objects. So a linear ordering L is polynomial time presented if the relation “ $z \in L$ ” is p -time in $|x|$. Consider $x \in \{0, 1\}^{<\omega}$. Cenzer and Remmel [29], Grigorieff [87], Nerode and Remmel [175], and others, have obtained very interesting results here. We will satisfy ourselves by merely quoting two results.

Theorem 8.4

- (i) (Cenzer and Remmel [29]) *There is a computable linear ordering of type $\omega + \omega$ not computably isomorphic to a p -time ordering.*
- (ii) (Remmel [198]) *For every infinite polynomial time subset of $\{0, 1\}^*$ and every recursive L linear ordering of the form $\omega + \zeta\lambda + \omega^*$, there exists a recursive linear ordering L' which is isomorphic to L but which is not recursively isomorphic to any polynomial time linear ordering whose universe is A .*

Theorem 8.5 (Grigorieff [87]) *Every computable linear ordering has a copy in $\text{DTIME-SPACE}(n, \log(n))$.*

We note that in proving Theorem 8.5, Grigorieff proved that if L is any recursive linear ordering which has a recursive L -increasing (L -decreasing) sequence which is either cofinal (co-initial) in L or has a limit in L , then L is recursively isomorphic to a linear ordering over the binary representation of the natural numbers which is in $\text{DTIME-SPACE}(n, \log(n))$. Thus Theorem 8.4(ii) is as strong as possible. Remmel [199] defined a natural notion of polynomial time categoricity relative to a polynomial time set $A \subseteq \{0, 1\}^*$ and showed that there are no polynomial time categorical linear orderings over the binary representation of the natural numbers or over the unary representation of the natural numbers. This is obviously an area of great potential.

The final one we will mention is an old one that was part of a major area of research in the 1960's: *constructive order types*. Dekker and Myhill asked what happens if we can only see comparability relations through “computable eyes”. Suppose we have a partial computable injective function f . Evidently from an effective point of view $\text{dom } f$ and $\text{ran } f$ are the “same size”.

But clearly f matches many other sets. For if $A \subseteq \text{dom } f$ then A is matched to $f(A)$. This leads to the following definition: we say that Q is *computably equivalent* to R ($Q \approx R$) if there is an injective partial computable function f with $\text{dom } f \supset Q$, $\text{ran } f \supset R$ and $f(Q) = R$. It is easy to see that $\approx$ is an equivalence relation and the equivalence classes are called *computable equivalence types* (CET's). If A is a set, let (A) denote the CET of A . CET's have been extensively studied by Dekker, Myhill, Barback, McLaughlin, Nerode and others. The reader is directed to McLaughlin [154] for details of this development. One thinks of computable equivalence as generating an analogue of the equipollence. We say $(A) \leq (B)$ iff there are representatives of A and B of (A) and (B) and a set C with $A \oplus C \approx B$. One of the key notions is that of an *isol*. We say that (A) is an *isol* if A is finite or immune. The isols are the essential building blocks of this system since A is an *isol* iff A is not computably equivalent to a proper subset of itself. So isols are the analogues of the Dedekind finite sets.

The major idea in the programme of investigating CET's was to try to understand what arithmetic truths of the integers would smoothly lift to the isols. For instance, we define $+$ and $\cdot$ in the obvious ways, and soon discover that for isols many classical combinatorial identities are true for the isols. This work is particularly interesting in view of the fact that we now know that the isols really are models of *choice free mathematics obtained via Kleene realisability* (c.f. McCarty [153]). Thus they are really strongly related to classical intuitionistic systems. In many ways, the culmination of much of the early work on isols were the beautiful results of Nerode [173] who extended some earlier work of Myhill to show how to lift a wide class (Horn sentences) of properties to the isols. Viewed nowadays, Nerode's argument is clearly a forcing construction (before Cohen, although not set theoretical forcing) and later Ellentuck [65] recast Nerode's work in terms of boolean valued models. So the CET work can be viewed as looking at preservation theorems for a certain class of forcing construction. The reader should look at Crossley [39] for a nice amount of the development and philosophy of some of the originators of the area. It is quite interesting to speculate on whether the methods or ideas can be used to capture a fragment of uniform (classical) combinatorics.

Crossley [36] and Manaster (unpublished) both began to look at similar analogues of the ordinals and were led to several definitions of "effective order type" (also Rice [200]). We look at only one (see e.g., Eisenberg and Remmel [64] for others). We say $A \approx B$ (A and B have the *same constructive*

order type) if $A, B \subseteq \mathbb{Q}$ and there is an injective partial computable f with $f(A) = B$ and such that if $f(x) \downarrow$ and $f(y) \downarrow$ then $x \leq y$. Again we let (A) denote the equivalence class, the COT, of A . An isolated COT is called *Dedekind*. Crossley [36] developed quite an elaborate theory of COT's and Dedekind COT's. For instance, he showed that

Theorem 8.6 (Crossley [36]) *If P is a unary nontrivial function built up from $+$, $\cdot$, exponentiation, taking converses and allowing Dedekind COT parameters, then for Dedekind COT's*

$$P(X) = P(Y) \quad \leftrightarrow \quad X = Y.$$

Crossley and Nerode [41] later pushed Theorem 8.6 a lot further (along the lines of Nerode [173]), and to much more general settings. Recently Nerode and Remmel [175] have looked at p -time versions of this work.

Finally one area we have not even looked at is *computable graph theory*. Here there is an extensive literature such as Manaster and Fosenstein's [148] computable analysis of P. Hall's marriage theorem. Other work can be found in, for instance, Aharoni, Magidor and Shore [1], Bean [17, 18], Biegel and Gasarch [19, 20, 21, 22], Gasarch [78], Gasarch, Kueker and Mount [80] Gasarch and Lockwood [81], Hirst [98], Jockusch [104], Kierstead [118, 119], Remmel [194, 195], Schmerl [216, 217, 218], Shore [221] and Tverberg [240]. Here we refer to Gasarch's survey in the present volume.

References

- [1] R. Aharoni, M. Magidor and R. A. Shore, On the strength of König's duality theorem for infinite bipartite graphs, *J. Combin. Theory, Ser. B*, **54** (1992) 257–290.
- [2] K. Ambos-Spies, S. B. Cooper, and S. Lempp, Initial segments of recursive linear orders, (to appear).
- [3] I. Anderson, *Combinatorics of Finite Sets*, Oxford Sci. Publ., Oxford Univ. Press, 1987.
- [4] C. J. Ash, Recursive labelling systems and stability of recursive structures in hyperarithmetical degrees, *Trans. Amer. Math. Soc.*, **298** (1986) 497–514; errata *ibid.*, **310** (1988) 851.

- [5] C. J. Ash, Stability of recursive structures in the arithmetical degrees, *Ann. Pure Appl. Logic*, **32** (1986) 113–135.
- [6] C. J. Ash, Categoricity in hyperarithmetical degrees, *Ann. Pure Appl. Logic*, **34** (1987) 1–14.
- [7] C. J. Ash, Labelling systems and r.e. structures, *Ann. Pure Appl. Logic*, **47** (1990) 99–119.
- [8] C. J. Ash, A construction for recursive linear orderings, *J. Symbolic Logic*, **56** (1991) 673–683.
- [9] C. J. Ash, C. G. Jockusch Jr. and J. F. Knight, Jumps of orderings, *Trans. Amer. Math. Soc.*, **319** (1990) 573–599.
- [10] C. J. Ash and J. F. Knight, Pairs of recursive structures, *Ann. Pure Appl. Logic*, **46** (1990) 211–234.
- [11] C. J. Ash and J. F. Knight, *Computable Structures and the Hyperarithmetical Hierarchy*, monograph, (to appear).
- [12] C. J. Ash, J. F. Knight, M. Manasse and T. A. Slaman, Generic copies of countable structures, *Ann. Pure Appl. Logic*, **42** (1989) 195–205.
- [13] C. J. Ash and A. Nerode, Intrinsically recursive relations, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981), 26–41.
- [14] K. Baker, D. C. Fishburn and F. Roberts, Partially ordered sets of dimension 2, *Networks*, **2** (1971) 11–28.
- [15] E. J. Barker, Intrinsically Σ_α^0 relations, *Ann. Pure Appl. Logic*, **39** (1988) 105–130.
- [16] G. Baumslag, G. E. Dyer and C. Miller III, On the integral homology of finitely presented groups, *Topology*, **22** (1983) 27–46.
- [17] D. R. Bean, Effective coloration, *J. Symbolic Logic*, **41** (1976) 469–480.
- [18] D. R. Bean, Recursive Euler and Hamilton paths, *Proc. Amer. Math. Soc.*, **55** (1976) 385–394.

- [19] R. Beigel and W. I. Gasarch, On the complexity of finding the chromatic number of a recursive graph I: The bounded case, *Ann. Pure Appl. Logic*, **45** (1989) 1–38.
- [20] R. Beigel and W. I. Gasarch, On the complexity of finding the chromatic number of a recursive graph II: The unbounded case, *Ann. Pure Appl. Logic*, **45** (1989) 227–246.
- [21] R. Beigel and W. I. Gasarch, The mapmaker's dilemma, *Discrete Appl. Math.*, **34** (1991) 37–48, [also in: Univ. Maryland Tech. Rep. TR 2262 (1989)].
- [22] R. Beigel and W. I. Gasarch, Applications of binary search to recursive graph theory, Univ. Maryland Tech. Rep. TR 1804.
- [23] R. Bonnet, Stratifications et extension des genres de chaînes dénombrables, *C. R. Acad. Sci. Paris*, **269** (1969) 880–882.
- [24] R. Bonnet and M. Pouzet, Extension et stratification d'ensembles dispersés, *C. R. Acad. Sci. Paris*, **268** (1969) 1512–1515.
- [25] R. Bonnet and M. Pouzet, Linear extensions of ordered sets, in: *Ordered Sets*, (Proc. NATO Adv. Study Inst., Banff, Alta, Aug. 28 – Sept. 12, 1981), I. Rival (ed.), NATO Adv. Study Inst. Ser. C: Math. Phys. Sci., (Reidel, Dordrecht and Boston), **83** (1982) 125–170.
- [26] W. W. Boone, Certain simple unsolvable problems in group theory, I–VI, *Nederl. Akad. Wetensch. Proc. Ser. A*, **57** (1954) 231–237, 492–497; **58** (1955) 252–256, 571–577; **60** (1957) 22–27, 227–232.
- [27] W. C. Calhoun, The Lattice of Ideas of Recursively Enumerable Degrees, Ph.D. Thesis, Univ. California, Berkeley, CA, (1992).
- [28] W. C. Calhoun, Incomparable prime ideals of recursively enumerable degrees, *Ann. Pure Appl. Logic*, **63** (1993) 39–56.
- [29] D. Cenzer and J. B. Remmel, Polynomial-time versus recursive models, *Ann. Pure Appl. Logic*, **54** (1991) 17–58.
- [30] C. C. Chang and H. J. Keisler, *Model Theory*, 3rd. edn., Stud. Logic Found. Math., **73** (1990); [1st. edn. 1973, 2nd. edn. 1977].

- [31] K. H. Chen, Recursive well founded orderings, *Ann. Math. Logic*, **13** (1978) 117–147.
- [32] J. Chisholm and M. Moses, Undecidable linear orderings that are n -recursive for each n , (in preparation).
- [33] P. A. Cholak, S. S. Goncharov, B. M. Khoussainov and R. A. Shore, On recursively categorical models, (in preparation).
- [34] R. Coles, R. G. Downey and B. M. Khoussainov, Initial segments of computable linear orderings, (to appear).
- [35] T. Craven, The Boolean space of orderings of a field, *Trans. Amer. Math. Soc.*, **209** (1975) 225–235.
- [36] J. N. Crossley, *Constructive Order Types*, (North-Holland, Amsterdam, 1969).
- [37] J. N. Crossley, Recursive equivalence, *Bull. London Math. Soc.*, **2** (1970) 129–151. (Survey paper)
- [38] J. N. Crossley, (ed.), *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981).
- [39] J. N. Crossley, *Reminiscences of Logicians II*, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981), 1–25.
- [40] J. N. Crossley, A. B. Manaster and M. Moses, Recursive categoricity and recursive stability, (Special issue: 2nd. Southeast Asian Logic Conf., Bangkok, 1984), *Ann. Pure Appl. Logic*, **31** (1986) 191–204.
- [41] J. N. Crossley and A. Nerode, Combinatorial Functors, *Ergeb. Math. Grenzgeb. Ser. 3*, **81** (1974).
- [42] J. N. Crossley and R. J. Parikh, On isomorphisms of recursive well-orderings, *J. Symbolic Logic*, **28** (1963) 308 (abstract).
- [43] M. D. Davis and E. J. Weyuker, *Computability, Complexity and Languages*, *Fundamentals of Theoretical Computer Science*, (Academic Press, New York, 1983), [2nd. edn. with R. Sigal, 1994].

- [44] M. Dehn, Über unendliche diskontinuierliche gruppen, *Math. Ann.*, **71** (1912) 116–144; [translated in: *Papers on group theory and topology*, translated from the German and with introductions and an appendix by J. Stillwell, with an appendix by O. Schreier, (Springer-Verlag, 1987), 133–178, with introduction 127–132].
- [45] P. Dehornoy, Turing complexity of the ordinals, *Inform. Process. Lett.*, **23** (1986) 167–170.
- [46] J. C. E. Dekker and J. Myhill, Recursive equivalence types, *Publ. Math. Univ. California*, **3** (1960) 67–214.
- [47] R. P. Dilworth, A decomposition theorem for partially ordered sets, *Ann. of Math.*, Ser. 2, **51** (1950) 161–166.
- [48] R. G. Downey, A Problem of Rosenstein, (unpublished manuscript, 1985)
- [49] R. G. Downey, Recursion theory and linear orderings, *Victoria Univ. Res. Rep.* 91–71, (June, 1991).
- [50] R. G. Downey, Every recursive Boolean algebra is isomorphic to one with incomplete atoms, *Ann. Pure Appl. Logic*, **60** (1993) 193–206.
- [51] R. G. Downey, On presentations of algebraic structures, in: *Complexity, Logic and Recursion Theory*, A. Sorbi, (ed.), *Lecture Notes in Pure and Appl. Math.*, **187** (1997) 157–205.
- [52] R. G. Downey and C. G. Jockusch Jr., T-degrees, jump classes and strong reducibilities, *Trans. Amer. Math. Soc.*, **301** (1987) 103–136.
- [53] R. G. Downey and C. G. Jockusch Jr., Every low Boolean algebra is isomorphic to a recursive one, *Proc. Amer. Math. Soc.*, **122** (1994) 871–880.
- [54] R. G. Downey and C. G. Jockusch Jr., Effective presentability of Boolean algebras of Cantor–Bendixson rank 1, *J. Symbolic Logic*, (to appear).
- [55] R. G. Downey and J. F. Knight, Orderings with α -th jump degree $\mathbf{0}^\alpha$, *Proc. Amer. Math. Soc.*, **114** (1992) 545–552.
- [56] R. G. Downey and S. A. Kurtz, Recursion theory and ordered groups, *Ann. Pure Appl. Logic*, **32** (1986) 137–151.

- [57] R. G. Downey and S. Lempp, On the proof theoretical strength of the Dushnik-Miller theorem for countable linear orderings, (in preparation).
- [58] R. G. Downey and M. F. Moses, On choice sets and strongly nontrivial self-embeddings of recursive linear orders, *Z. Math. Logik Grundlag. Math.*, **35** (1989) 237–246.
- [59] R. G. Downey and M. Moses, Recursive linear orders with incomplete successivities, *Trans. Amer. Math. Soc.*, **326** (1991) 653–668.
- [60] B. Dushnik and E. W. Miller, Concerning similarity transformations of linearly ordered sets, *Bull. Amer. Math. Soc.*, **46** (1940) 322–326.
- [61] B. Dushnik and E. W. Miller, Partially ordered sets, *Amer. J. Math.*, **63** (1941) 600–610.
- [62] V. D. Dzgoev and S. S. Goncharov, Autostability of models, *Algebra i Logika*, **19** (1980) 45–58, 132; [translated in: *Algebra and Logic*, **19** (1980) 28–37].
- [63] E. F. Eisenberg, Effective Isomorphisms of Algebraic Systems, Ph.D. Thesis, Cornell Univ., Ithaca, NY, (1974).
- [64] E. F. Eisenberg and J. B. Remmel, Effective isomorphisms of algebraic structures, in: *Patras Logic Symposion*, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) G. Metakides (ed.), *Stud. Logic Found. Math.*, **109** (1982) 95–122.
- [65] E. Ellentuck, The first order properties of Dedekind finite integers, *Fund. Math.*, **63** (1968) 7–25.
- [66] L. J. Feiner, Orderings and Boolean Algebras not Isomorphic to Recursive Ones, Ph.D. Thesis, MIT, Cambridge, MA, (1967).
- [67] L. J. Feiner, Hierarchies of Boolean algebras, *J. Symbolic Logic*, **35** (1970) 365–374.
- [68] L. J. Feiner, The strong homogeneity conjecture, *J. Symbolic Logic*, **35** (1970) 375–377.
- [69] L. J. Feiner, Degrees of non-recursive presentability, *Proc. Amer. Math. Soc.*, **38** (1973) 621–624.

- [70] A. Feldman, Recursion theory in a lower semilattice, *J. Symbolic Logic*, **57** (1992) 892–911.
- [71] S. Fellner, Recursive and Finite Axiomatizability of Linear Orderings, Ph.D. Thesis, Rutgers Univ., New Brunswick, NJ, (1976).
- [72] R. M. Friedberg, Two recursively enumerable sets of incomparable degrees of unsolvability, *Proc. Nat. Acad. Sci. U.S.A.*, **43** (1957) 236–238.
- [73] R. M. Friedberg, Three theorems on recursive enumeration: I decomposition, II maximal set, III enumeration without duplication, *J. Symbolic Logic*, **23** (1958) 309–316.
- [74] R. M. Friedberg and H. Rogers, Reducibility and Completeness for sets of integers, *Z. Math. Logik Grundlag. Math.*, **5** (1959) 117–125.
- [75] H. M. Friedman and J. L. Hirst, Weak comparability of well orderings and reverse mathematics, *Ann. Pure Appl. Logic*, **47** (1990) 11–29.
- [76] H. M. Friedman, S. G. Simpson and R. L. Smith, Countable algebra and set existence axioms, *Ann. Pure Appl. Logic*, **25** (1983) 141–181; addendum, *ibid.* **27** (1985) 319–320.
- [77] L. Fuchs, Infinite abelian groups, Vol. 2, *Pure Appl. Math.*, **36–II** (1973).
- [78] W. I. Gasarch, A hierarchy of functions with applications to recursive graph theory, Univ. Maryland Tech. Rep. TR 1651.
- [79] W. I. Gasarch, D. W. Keuter and D. M. Mount, Highly recursive rooted graphs, *Congr. Numer.*, **69** (1989) 97–102; [also in: *Proc. 19th. S. E. Conf. Combinatorics*, Baton Rouge, LA, Feb. 15–19, 1988, F. Hoffman, R. C. Mullin, R. G. Stanton and K. B. Reid, (eds.), *Congr. Numer.*, **67** (1988)], also in: *Proc. 18th. Manitoba Conf. Numer. Math. and Comput.*, Winnipeg, MB, Sept. 29 – Oct. 1, 1988), D. S. Meek, R. G. Stanton and G. H. J. van Rees, (eds.) *Congr. Numer.*, **69** (1989)].
- [80] W. I. Gasarch, D. W. Keuter and D. M. Mount, Recursive categoricity of highly recursive graphs, (preprint).
- [81] W. I. Gasarch and M. Lockwood, The existence of matchings for recursive and highly recursive bipartite graphs, Univ. Maryland Tech. Rep. TR 2029.

- [82] S. S. Goncharov, The constructivizability of superatomic Boolean algebras (Russian), *Algebra i Logika*, **12** (1973) 31–40, 120; [translated in: *Algebra and Logic*, **12** (1973) 17–22].
- [83] S. S. Goncharov, Certain properties of the constructivization of Boolean algebras (Russian), *Sibirsk. Math. Zh.*, **16** (1975) 264–278; [translated in: *Siberian Math. J.*, **16** (1975) 203–214].
- [84] S. S. Goncharov, The quantity of nonautoequivalent constructivizations (Russian), *Algebra i Logika*, **16** (1977) 257–282, 377; [translated in: *Algebra and Logic*, **16** (1977) 169–185].
- [85] S. S. Goncharov, Autostability of models and abelian groups (Russian), *Algebra i Logika*, **19** (1980) 23–44; [translated in: *Algebra and Logic*, **19** (1980) 13–27].
- [86] S. S. Goncharov and A. T. Nurtazin, Constructive models of complete decidable theories (Russian), *Algebra i Logika*, **12** (1973) 125–142, 243; [translated in: *Algebra and Logic*, **12** (1973) 67–77].
- [87] S. Grigorieff, Every recursive linear ordering has a copy in $\text{DTIME-SPACE}(n, \log(n))$, *J. Symbolic Logic*, **55** (1990) 260–276.
- [88] V. S. Harizanov, Some effects of Ash-Nerode and other decidability conditions on degree spectra, *Ann. Pure Appl. Logic*, **55** (1991) 51–65.
- [89] V. S. Harizanov, Uncountable degree spectra, *Ann. Pure Appl. Logic*, **54** (1991) 255–263.
- [90] V. S. Harizanov, Uncountable degree spectra, *Ann. Pure Appl. Logic*, **54** (1991) 255–263.
- [91] L. Harrington, Building non-standard models of Peano arithmetic, (handwritten notes, 1979).
- [92] L. S. Hay, A. B. Manaster and J. G. Rosenstein, Concerning partial recursive similarity transformations of linearly ordered sets, *Pacific J. Math.*, **71** (1977) 57–70.
- [93] A. G. Higman, Subgroups of finitely presented groups, *Proc. Roy. Soc. London Ser. A*, **262** (1961) 455–475.+

- [94] A. G. Higman and E. Scott, *Existentially Closed Groups*, London Math. Soc. Monographs (N.S.), **3**, (Oxford Univ. Press, 1988).
- [95] T. Hiraguchi, On the dimension of orders, *Sci. Rep. Kanazawa Univ.*, **4** (1955) 1–20.
- [96] G. Hird, *Recursive Properties of Relations on Models*, Ph.D. Thesis, Monash Univ., Clayton, Victoria, Australia, (1983).
- [97] G. Hird, Recursive properties of intervals of recursive linear orders, in: *Logical Methods*, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), *Progr. Comput. Sci. Appl. Logic*, **12** (1993) 422–437.
- [98] J. L. Hirst, Marriage theorems and reverse mathematics, in: *Logic and Computation*, (Proc. Workshop, CMU, Pittsburg, June 30 – July 2, 1987), W. Sieg, (ed.), *Contemp. Math.*, **106** (1990) 181–196.
- [99] O. Hölder, Die Axiome der Quantität und die Lehre vom Mass, *Ber. Verhand. König. Sächs. Ges. Wiss. Leipzig, Math. Phys. Cl.*, **53** (1901) 1–64.
- [100] L. Hopkins, *Some Problems Involving Combinatorial Structures Determined by Intersection of Intervals*, Ph.D. Thesis, Univ. South Carolina, Columbia, SC, (1981).
- [101] C. G. Jockusch Jr., *Reducibilities in Recursive Function Theory*, Ph.D. Thesis, MIT, Cambridge, MA, (1966).
- [102] C. G. Jockusch Jr., Semirecursive sets and positive reducibility, *Trans. Amer. Math. Soc.*, **131** (1968) 420–436.
- [103] C. G. Jockusch Jr., Relationships between reducibilities, *Trans. Amer. Math. Soc.*, **142** (1969) 229–237.
- [104] C. G. Jockusch Jr., Ramsey's theorem and recursion theory, *J. Symbolic Logic*, **37** (1972) 268–280.
- [105] C. G. Jockusch Jr. and R. I. Soare, Degrees of members of Π_1^0 classes, *Pacific J. Math.*, **40** (1972) 605–616.
- [106] C. G. Jockusch Jr. and R. I. Soare, Π_1^0 classes and degrees of theories, *Trans. Amer. Math. Soc.*, **173** (1972) 33–56.

- [107] C. G. Jockusch Jr. and R. I. Soare, Degrees of orderings not isomorphic to recursive linear orderings, (Special issue: Internatl. Sympos. Math. Logic and its Applications, Nagoya, 1988), *Ann. Pure Appl. Logic*, **52** (1991) 39–64.
- [108] C. G. Jockusch Jr. and R. I. Soare, Boolean algebras, Stone spaces, and the iterated Turing jump, *J. Symbolic Logic*, **59** (1994) 1121–1138.
- [109] D. Kelly and W. T. Trotter Jr., Dimension theory for ordered sets, in: *Ordered Sets*, (Proc. NATO Adv. Study Inst., Banff, Alta, Aug. 28 – Sept. 12, 1981), I. Rival (ed.), NATO Adv. Study Inst. Ser. C: Math. Phys. Sci., (Reidel, Dordrecht and Boston), **83** (1982) 171–212.
- [110] J. Ketonen, The structure of countable Boolean algebras, *Ann. of Math.*, Ser. 2, **108** (1978) 41–89.
- [111] N. G. Khisamiev, Connection between constructivizability and strong constructivizability for different classes of abelian groups (Russian), *Algebra i Logika*, **23** (1984) 319–335; [translated in: *Algebra and Logic*, **23** (1984) 220–233].
- [112] N. G. Khisamiev, Hierarchies of torsion-free abelian groups (Russian), *Algebra i Logika*, **25** (1986) 205–226, 244; [translated in: *Algebra and Logic*, **25** (1986) 128–142].
- [113] Z. G. Khisamiev and N. G. Khisamiev, Nonconstructivizability of the reduced part of a strongly constructive torsion-free abelian group (Russian), *Algebra i Logika*, **24** (1985) 108–118, 123; [translated in: *Algebra and Logic*, **24** (1985) 69–76].
- [114] B. M. Khossainov and A. Nerode, Automatic presentations of structures, in: *Logic and Computational Complexity*, (Papers from Internatl. Workshop, LCC '94, Oct. 13–16, 1994, Indianapolis, IN), D. Leivant (ed.), *Lecture Notes in Comput. Sci.*, **960** (1995) 367–392.
- [115] B. M. Khossainov and A. Nerode, *Automatic Model Theory*, (monograph in preparation).
- [116] B. M. Khossainov, A. Nies and R. A. Shore, Computable models of theories with few models, *Notre Dame J. Formal Logic*, **38** (1997) 165–178.

- [117] H. A. Kierstead, An effective version of Dilworth's theorem, *Trans. Amer. Math. Soc.*, **268** (1981) 63–77.
- [118] H. A. Kierstead, Recursive colorings of highly recursive graphs, *Canad. J. Math.*, **33** (1981) 1279–1290.
- [119] H. A. Kierstead, An effective version of Hall's theorem, *Proc. Amer. Math. Soc.*, **88** (1983) 124–128.
- [120] H. A. Kierstead, Recursive ordered sets, in: *Combinatorics and Ordered Sets*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf., Humboldt State Univ., Arcata, CA, Aug. 11–17, 1985), I. Rival, (ed.), *Contemp. Math.*, **57** (1986) 75–102.
- [121] H. A. Kierstead, On Π_1 -automorphisms of recursive linear orders, *J. Symbolic Logic*, **52** (1987) 681–688.
- [122] H. A. Kierstead, G. McNulty and W. T. Trotter Jr., A theory of recursive dimension for ordered sets, *Order*, **1** (1984) 67–82.
- [123] H. A. Kierstead and W. T. Trotter Jr., An extremal problem in recursive combinatorics, in: *Combinatorics, Graph Theory and Computing*, Vol. 2, (Proc. 12th. S. E. Conf., Baton Rouge, LA, 1981, *Congr. Numer.*, **33** (1981) 143–153.
- [124] J. F. Knight, Degrees coded into jumps of orderings, *J. Symbolic Logic*, **51** (1986) 1034–1042.
- [125] J. F. Knight, Effective construction of models, in: *Logic Colloquium '84*, (Manchester, July 15–24, 1984), J. B. Paris, A. J. Wilkie and G. M. Wilmers, (eds.), *Stud. Logic Found. Math.*, **120**, (1986) 105–119.
- [126] J. F. Knight, Degrees of models with prescribed Scott set, in: *Classification Theory*, (Proc. Joint U.S.–Israel Workshop on Model Theory in Math. Logic, Chicago, Dec. 15–19, 1985), J. T. Baldwin, (ed.), *Lecture Notes in Math.*, **1292** (1987) 182–191.
- [127] J. F. Knight, Constructions by transfinitely many workers, *Ann. Pure Appl. Logic*, **48** (1990) 237–259.
- [128] J. F. Knight, A metatheorem for constructions by finitely many workers, *J. Symbolic Logic*, **55** (1990) 787–804.

- [129] J. F. Knight, Coding a family of sets, (Special issue: Computability Theory, Proc. Conf., Oberwolfach 1996), *Ann. Pure Appl. Logic*, (to appear).
- [130] A. I. Kokorin and V. M. Kopytov, Fully Ordered Groups, translated from Russian by D. Louvish, (Halsted Press, John Wiley and Sons, 1974).
- [131] G. Kreisel, J. R. Shoenfield and H. Wang, Number theoretic concepts and recursive well-orderings, *Arch. Math. Logik Grundlag.*, **5** (1960) 42–64.
- [132] J. Kruskal, Well-quasi-ordering, the tree theorem and Vazsonyi's Conjecture, *Trans. Amer. Math. Soc.*, **95** (1960) 210–225.
- [133] A. H. Lachlan, Lower bounds for pairs of recursively enumerable degrees, *Proc. London Math. Soc.*, Ser. 3, **16** (1966) 537–569.
- [134] C. H. Langford, Some theorems on deducibility, *Ann. of Math.*, Ser. 2, **28** (1926/27) 16–40.
- [135] P. La Roche, Recursively presented Boolean algebras, *Notices Amer. Math. Soc.*, **24:6** (1977) A 552–553 (abstract).
- [136] R. Laver, On Fraïssé's order type conjecture, *Ann. of Math.*, **93** (1971) 89–111.
- [137] S. Lempp and M. Lerman, Priority arguments using iterated trees of strategies, in: *Recursion Theory Week*, (Proc. Conf., Oberwolfach, Mar. 19–25, 1989), K. Ambos-Spies, G. H. Müller and G. E. Sachs, (eds.), *Lecture Notes in Math.*, **1432** (1990) 277–296.
- [138] M. Lerman, On recursive linear orderings, in: *Logic Year 1979–1980*, (Proc. Seminars and Conf. Math. Logic, Univ. Connecticut, Storrs, CT, 1979/80), M. Lerman, J. H. Schmerl and R. I. Soare, (eds.), *Lecture Notes in Math.*, **859** (1981) 132–142.
- [139] M. Lerman, Degrees of Unsolvability, *Local and Global Theory*, *Perspect. in Math. Logic*, (1983).
- [140] M. Lerman and J. G. Rosenstein, Recursive linear orderings, in: *Patras Logic Symposion*, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) G. Metakides (ed.), *Stud. Logic Found. Math.*, **109** (1982) 123–136.
- [141] M. Lerman and J. H. Schmerl, Theories with recursive models, *J. Symbolic Logic*, **44** (1979) 59–76.

- [142] M. Lerman, J. H. Schmerl and R. I. Soare (eds.), *Logic Year 1979-1980*, (Proc. Seminars and Conf. Math. Logic, Univ. Connecticut, Storrs, CT, 1979/80), *Lecture Notes in Math.*, 859 (1981).
- [143] C. R. Lin, *Recursion Theory on Countable Abelian Groups*, Ph.D. Thesis, Cornell Univ., Ithaca, NY, (1977).
- [144] C. R. Lin, The effective content of Ulm's Theorem, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1-4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981) 147-160.
- [145] J. M. MacIntyre, Transfinite extensions of Friedberg's completeness criterion, *J. Symbolic Logic*, **42** (1977) 1-10.
- [146] A. B. Manaster and J. B. Remmel, Partial orderings of fixed finite dimension: model companions and density, *J. Symbolic Logic*, **46** (1981) 789-802.
- [147] A. B. Manaster and J. B. Remmel, Some recursion theoretic aspects of dense two-dimensional partial orderings, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1-4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981) 161-188.
- [148] A. B. Manaster and J. G. Rosenstein, Effective matchmaking (recursion theoretical aspects of a theorem of Philip Hall, *Proc. London Math. Soc.*, Ser. 3, **25** (1972) 615-654.
- [149] A. B. Manaster and J. G. Rosenstein, Two-dimensional partial orderings: recursive model theory, *J. Symbolic Logic*, **45** (1980) 121-132.
- [150] D. Marker, Degrees of models of true arithmetic, in: *Logic Colloq. '81*. (Proc. Herbrand Sympos., Marseilles, July, 16-24, 1981), J. Stern (ed.), *Stud. Logic Found. Math.*, **107** (1982) 233-242.
- [151] D. A. Martin, Classes of recursively enumerable sets and degrees of unsolvability, *Z. Math. Logik Grundlag. Math.*, **12** (1966) 295-310.
- [152] Yu. V. Matijasevič, Enumerable sets are diophantine (Russian), *Dokl. Akad. Nauk SSSR*, **191** (1970) 279-282; [translated in: *Soviet Math. - Dokl.*, **11** (1970) 354-357].

- [153] C. McCarty, Realizability and recursive set theory, *Ann. Pure Appl. Logic*, **32** (1986) 153–183.
- [154] T. G. McLaughlin, Regressive Sets and the Theory of Isols, *Lecture Notes in Pure and Appl. Math.*, **66** (1982).
- [155] G. F. McNulty, Infinite ordered sets, a recursive perspective, in: *Ordered Sets*, (Proc. NATO Adv. Study Inst., Banff, Alta, Aug. 28 – Sept. 12, 1981), I. Rival (ed.), NATO Adv. Study Inst. Ser. C: Math. Phys. Sci., (Reidel, Dordrecht and Boston), **83** (1982) 299–330.
- [156] J. Mead, Recursive prime models for Boolean algebras, *Colloq. Math.*, **41** (1979) 25–33.
- [157] G. Metakides, (ed.), Patras Logic Symposium, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) *Stud. Logic Found. Math.*, **109** (1982).
- [158] G. Metakides and A. Nerode, Recursion theory and algebra, in: *Algebra and Logic*, (Proc. 14th. Summer Res. Inst. Austral. Math. Soc., Monash Univ., Clayton, Vic., Australia, Jan. 6 – Feb. 16, 1974), J. N. Crossley, (ed.), *Lecture Notes in Math.*, **450** (1975) 209–219.
- [159] G. Metakides and A. Nerode, Recursively enumerable vector spaces, *Ann. Math. Logic*, **11** (1977) 147–171.
- [160] G. Metakides and A. Nerode, Effective content of field theory, *Ann. Math. Logic*, **17** (1979) 289–320.
- [161] G. Metakides and J. B. Remmel, Recursion theory on orderings I, A model theoretic setting, *J. Symbolic Logic*, **44** (1979) 383–402.
- [162] M. Minsky, Recursive unsolvability of Post’s problem of “tag” and other topics in the theory of Turing machines, *Ann. of Math.*, Ser. 2, **74** (1961) 437–455.
- [163] L. Mirsky, A dual to Dilworth’s decomposition theorem, *Amer. Math. Monthly*, **78** (1971) 876–877.
- [164] A. S. Morozov, Countable homogeneous Boolean algebras (Russian), *Algebra i Logika*, **21** (1982) 269–282; [translated in: *Algebra and Logic*, **21** (1982) 181–190.

- [165] A. S. Morozov, Strong constructivizability of countable saturated Boolean algebras (Russian), *Algebra i Logika*, **21** (1982) 193–203; [translated in: *Algebra and Logic*, **21** (1982) 130–137.
- [166] A. S. Morozov, Groups of Computable Automorphisms, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 311–345.
- [167] M. Moses, Recursive Properties of Isomorphism Types, Ph.D. Thesis, Monash Univ., Clayton, Victoria, Australia, (1983).
- [168] M. Moses, Recursive linear orders with recursive successivities, *Ann. Pure Appl. Logic*, **27** (1984) 253–264.
- [169] M. Moses, Relations intrinsically recursive in linear orders, *Z. Math. Logik Grundlag. Math.*, **32** (1986) 467–472.
- [170] M. Moses, Decidable discrete linear orders, *J. Symbolic Logic*, **53** (1988) 531–539.
- [171] M. Moses, n -recursive linear orders without $(n + 1)$ -recursive copies, in: *Logical Methods*, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), *Progr. Comput. Sci. Appl. Logic*, **12** (1993) 572–592.
- [172] A. A. Mučnik, On the unsolvability of the problem of reducibility in the theory of algorithms (Russian), *Dokl. Akad. Nauk SSSR*, **108** (1956) 194–197.
- [173] A. Nerode, Extensions to isols, *Ann. of Math.*, **73** (1961) 362–403.
- [174] A. Nerode and J. B. Remmel, Complexity-theoretic algebra II: Boolean algebras, (Special issue: 3rd. Asian Conf. Math. Logic, Beijing, Oct. 26–30, 1987), D. P. Yang, (ed.), *Ann. Pure Appl. Logic*, **44** (1989) 71–99.
- [175] A. Nerode and J. B. Remmel, Polynomial time equivalence types, in: *Logic and Computation*, (Proc. Workshop, CMU, Pittsburg, June 30 – July 2, 1987), W. Sieg, (ed.), *Contemp. Math.*, **106** (1990) 221–249.
- [176] B. H. Neumann, On ordered groups, *Amer. J. Math.*, **71** (1949) 1–18,

- [177] B. H. Neumann, Embedding theorems for ordered groups, *J. London Math. Soc.*, **35** (1960) 503–512.
- [178] P. S. Novikov, On the algorithmic unsolvability of the word problem in group theory (Russian), *Trudy Mat. Inst. Steklov.*, **44** (1955).
- [179] A. T. Nurtazin, Strong and weak constructivizations and computable families (Russian), *Algebra i Logika*, **13** (1974) 311–323, 364; [translated in: *Algebra and Logic*, **13** (1974) 177–184].
- [180] S. Oates, Jump Degrees of Groups, Ph.D. Thesis, Univ. Notre Dame, IN, (1989).
- [181] P. Odifreddi, Classical Recursion Theory. The Theory of Functions and Sets of Natural Numbers, *Stud. Logic Found. Math.*, **125** (1989).
- [182] J. Paris and L. Harrington, A mathematical incompleteness in Peano arithmetic, in: J. Barwise, (ed.), *Handbook of Mathematical Logic*, *Stud. Logic Found. Math.*, **90** (1977) 1133–1142.
- [183] M. G. Peretyat'kin, Every recursively enumerable extension of a theory of linear order has a constructive model (Russian), *Algebra i Logika*, **12** (1973) 211–219, 244; [translated in: *Algebra and Logic*, **12** (1973) 120–124].
- [184] A. G. Pinus, Efficient linear orders, *Siberian Math. J.*, **16** (1975) 956–962.
- [185] E. L. Post, Recursively enumerable sets of positive integers and their decision problems, *Bull. Amer. Math. Soc. (N.S.)*, **50** (1944) 284–316.
- [186] M. O. Rabin, Computable algebra, general theory and theory of computable fields, *Trans. Amer. Math. Soc.*, **95** (1960) 341–360.
- [187] M. Raw, Ph.D. Thesis, Univ. Wisconsin, Madison, WI, (1995).
- [188] J. B. Remmel, Recursively enumerable Boolean algebras, *Ann. Math. Logic*, **15** (1978) 75–107.
- [189] J. B. Remmel, Recursion theory on algebraic structures with independent sets, *Ann. Math. Logic*, **18** (1980) 153–191.
- [190] J. B. Remmel, Recursion theory on orderings II, *J. Symbolic Logic*, **45** (1980) 317–333.

- [191] J. B. Remmel, Recursive Boolean algebras with recursive atoms, *J. Symbolic Logic*, **46** (1981) 595–616.
- [192] J. B. Remmel, Recursive isomorphism types of recursive Boolean algebras, *J. Symbolic Logic*, **46** (1981) 572–594.
- [193] J. B. Remmel, Recursively categorical linear orderings, *Proc. Amer. Math. Soc.*, **83** (1981) 387–391.
- [194] J. B. Remmel, On the effectiveness of the Schröder-Bernstein theorem, *Proc. Amer. Math. Soc.*, **83** (1981) 379–386.
- [195] J. B. Remmel, Graph colorings and recursively bounded Π_1^0 -classes, *Ann. Pure Appl. Logic*, **32** (1986) 185–194.
- [196] J. B. Remmel, Recursively rigid Boolean algebras, *Ann. Pure Appl. Logic*, **36** (1987) 39–52.
- [197] J. B. Remmel, Recursive Boolean algebras, in: *Handbook of Boolean Algebras*, Vol. 3, J. D. Monk, R. Bonnet and S. Koppelberg, (eds.), (North-Holland, Amsterdam, NY, 1989), Ch. 25, 1097–1165.
- [198] J. B. Remmel, When is every recursive linear ordering of type μ recursively isomorphic to a polynomial time linear ordering over the natural numbers in binary form?, in: *Feasible Mathematics*, (Proc. Workshop, June 16–18, 1989, Cornell Univ. Ithaca, NY), S. Buss and P. J. Scott (eds.), *Progr. Comput. Sci. Appl. Logic*, **9** (1990) 321–350.
- [199] J. B. Remmel, Polynomial time categoricity and linear orderings, in: *Logical Methods*, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), *Progr. Comput. Sci. Appl. Logic*, **12** (1993) 713–746.
- [200] H. G. Rice, Recursive and recursively enumerable orders, *Trans. Amer. Math. Soc.*, **83** (1956) 277–300.
- [201] L. J. Richter, Degrees of Unsolvability of Models, Ph.D. Thesis, Univ. Illinois at Urbana-Champaign, Urbana, IL, (1977).
- [202] L. J. Richter, On automorphisms of the degrees that preserve jumps, *Israel J. Math.*, **32** (1979) 27–31.

- [203] L. J. Richter, Degrees of structures, *J. Symbolic Logic*, **46** (1981) 723–731.
- [204] I. Rival (ed.) *Ordered Sets*, (Proc. NATO Adv. Study Inst., Banff, Alta, Aug. 28 – Sept. 12, 1981), I. Rival (ed.), NATO Adv. Study Inst. Ser. C: Math. Phys. Sci., (Reidel, Dordrecht and Boston), **83** (1982).
- [205] I. Rival (ed.), *Graphs and Order*, The role of graphs in the theory of ordered sets and its applications, (Proc. NATO Adv. Study Inst., Banff, Alta., May 18–31, 1984), NATO Adv. Study Inst. Ser. C: Math. Phys. Sci., (Reidel, Dordrecht and Boston), **147** (1985).
- [206] I. Rival (ed.), *Combinatorics and Ordered Sets*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf., Humboldt State Univ., Arcata, CA, Aug. 11–17, 1985), *Contemp. Math.*, **57** (1986).
- [207] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, (1st. edn., McGraw-Hill, New York-Toronto, Ont.-London, 1967; 2nd. edn., MIT Press, Cambridge, Mass., London, 1987).
- [208] J. G. Rosenstein, Linear orderings, *Pure Appl. Math.*, **98** (1982).
- [209] J. G. Rosenstein, Recursive linear orderings, in: *Orders: Description and Roles*, *Ann. Discrete Math.*, **23** (1984) 465–475.
- [210] D. K. Roy, R.e. presented linear orders, *J. Symbolic Logic*, **48** (1983) 369–376.
- [211] D. K. Roy, Linear order types of nonrecursive presentability, *Z. Math. Logik Grundlag. Math.*, **31** (1985) 495–501.
- [212] D. K. Roy and R. Watnick, Finite condensations of recursive linear orders, *Studia Logika*, **47** (1988) 311–317.
- [213] G. E. Sacks, The recursively enumerable degrees are dense, *Ann. of Math.*, **80** (1964) 300–312.
- [214] G. E. Sacks, *Higher Recursion Theory*, *Perspect. in Math. Logic*, (1990).
- [215] A. Salomaa, *Computation and Automata*, (Cambridge Univ. Press, 1985).
- [216] J. H. Schmerl, Recursive colorings of graphs, *Canad. J. Math.*, **32** (1980) 821–830.

- [217] J. H. Schmerl, The effective version of Brooks' theorem, *Canad. J. Math.*, **34** (1982) 1036–1046.
- [218] J. H. Schmerl, Recursion theoretic aspects of graphs and orders, in: *Graphs and Order, The role of graphs in the theory of ordered sets and its applications*, (Proc. NATO Adv. Study Inst., Banff, Alta., May 18–31, 1984), I. Rival, (ed.), NATO Adv. Study Inst. Ser. C: Math. Phys. Sci., (Reidel, Dordrecht and Boston), **147** (1985) 467–484.
- [219] S. Schwarz, Quotient lattices, index sets and recursive linear orderings, Ph.D. Thesis, Chicago Univ., Chicago, IL, (1982).
- [220] S. Schwarz, Recursive automorphisms of recursive linear orderings, *Ann. Pure Appl. Logic*, **26** (1984) 69–73.
- [221] R. A. Shore, On the strength of Fraïssé's conjecture, in: *Logical Methods*, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), *Progr. Comput. Sci. Appl. Logic*, **12** (1993) 782–813.
- [222] S. G. Simpson, (ed.), *Logic and Combinatorics*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf. Applications. Math. Logic to Finite Combinatorics, Humboldt State Univ., Arcata, CA, Aug. 4–10, 1985), *Contemp. Math.*, **65** (1987).
- [223] S. G. Simpson, Unprovable theorems and fast-growing functions, in: *Logic and Combinatorics*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf. Applications Math. Logic to Finite Combinatorics, Humboldt State Univ., Arcata, CA, Aug. 4–10, 1985), S. G. Simpson, (ed.), *Contemp. Math.*, **65** (1987) 359–394.
- [224] S. G. Simpson, Subsystems of Second Order Arithmetic, (in preparation).
- [225] T. A. Slaman and W. H. Woodin, Extending partial orders to dense linear orders, (to appear).
- [226] R. L. Smith, Two theorems on autostability in p -groups, in: *Logic Year 1979–1980*, (Proc. Seminars and Conf. Math. Logic, Univ. Connecticut, Storrs, CT, 1979/80), M. Lerman, J. H. Schmerl and F. I. Soare, (eds.), *Lecture Notes in Math.*, **859** (1981) 302–311.

- [227] R. I. Soare, Constructive order types on cuts, *J. Symbolic Logic*, **34** (1969) 285–289.
- [228] R. I. Soare, Recursion theory and Dedekind cuts, *Trans. Amer. Math. Soc.*, **140** (1969) 271–294.
- [229] R. I. Soare, Automorphisms of the lattice of recursively enumerable sets, part I: maximal sets, *Ann. of Math.*, **100** (1974) 80–120.
- [230] R. I. Soare, The infinite injury priority method, *J. Symbolic Logic*, **41** (1976) 513–530.
- [231] R. I. Soare, Computational complexity, speedable and levelable sets, *J. Symbolic Logic*, **42** (1977) 545–563.
- [232] R. I. Soare, Recursively enumerable sets and degrees, *Bull. Amer. Math. Soc. (N.S.)*, **84** (1978) 1149–1181.
- [233] R. I. Soare, Computability and recursion, *Bulletin of Symbolic Logic*, **2** (1996) 284–321.
- [234] D. R. Solomon, Reverse Mathematics and Ordered Groups, Ph.D. Thesis, Cornell Univ., Ithaca, NY, (1998).
- [235] W. Szmielew, Elementary properties of Abelian groups, *Fund. Math.*, **41** (1955) 203–271.
- [236] E. Szpilrajn, (E. Marczewski), Sur l’extension de l’ordre partiel, *Fund. Math.*, **16** (1930) 386–389.
- [237] J. Thurber, Degrees of Boolean Algebras, Ph.D. Thesis, Univ. Notre Dame, Notre Dame, IN, (1994).
- [238] W. T. Trotter Jr. and J. I. Moore Jr., Characterization problems for graphs: partially ordered sets, lattices and families of sets, *Discrete Math.*, **16** (1976) 361–381.
- [239] W. T. Trotter Jr. and T. R. Monroe, A combinatorial problem involving graphs and matrices, *Discrete Math.*, **39** (1982) 87–101.
- [240] H. Tverberg, On Schmerl’s effective version of Brooks’s theorem, *J. Combin. Theory, Ser. B*, **37** (1984) 27–30.

- [241] R. Watnick, *Recursive and Constructive Linear Orderings*, Ph.D. Thesis, Rutgers Univ., New Brunswick, NJ, (1980).
- [242] R. Watnick, Constructive and recursive scattered order types, in: *Logic Year 1979-1980*, (Proc. Seminars and Conf. Math. Logic, Univ. Connecticut, Storrs, CT, 1979/80), M. Lerman, J. H. Schmerl and R. I. Soare, (eds.), *Lecture Notes in Math.*, **859** (1981) 312–326.
- [243] R. Watnick, A generalization of Tennenbaum's Theorem on effectively finite recursive linear orderings, *J. Symbolic Logic*, **49** (1984) 563–569.
- [244] C. E. M. Yates, A minimal pair of recursively enumerable degrees, *J. Symbolic Logic*, **31** (1966) 159–168.
- [245] P. R. Young, Linear orderings under one-one reducibility, *J. Symbolic Logic*, **31** (1966) 70–85.

Rod Downey
Mathematics Department
Victoria University
PO Box 600
Wellington
New Zealand
e-mail: 1rod.downey@vuw.ac.nz

Chapter 15

Computable Algebras and Closure Systems: Coding Properties

R. G. Downey and J. B. Remmel*

Contents

- Introduction
- 1. Preliminaries
- 2. Coding results
- 3. Sacks' splittings
- 4. Another example: $K(V_\infty)$
- 5. Conclusions and open questions
- References

Introduction

To show that some process in mathematics is effective (or computable) essentially requires one to explicitly give an algorithm for the process in question. On the other hand, some processes in mathematics are not algorithmic. The major common tool used in showing a process is *not* algorithmic is recursion theory. In fact, the traditional method of applying recursion theory to show that a problem is not computable is *coding*, which is, roughly speaking, the following procedure. We have a process in general mathematics we wish to show is non-computable. For example, in Hilbert's 10-th problem, we wish to show that there is no algorithm which applied to a diophantine equation will determine if it has an integer solution. To do this we build an r.e. *set* and then (somehow) *can code this set* into the problem. For example, Matijasevic [60, 59] showed that any r.e. set could be the set of solutions to a diophantine

*First author partially supported by grant (NUS)RP-85/83, the second by NSF grant DMS-9306427 and Dept. of Commerce Agreement 70-NANB5H1164. First author supported in part by the Marsden Fund for basic Science in New Zealand.

equation. Then since there is no algorithm to decide if an r.e. set is empty, there is no algorithm, which applied to any diophantine equation, will determine if it has a solution. Similarly, we may want to show that it is not the case that every recursive k -colorable graph has a recursive k -coloring. We show that given any pair of r.e. sets A and B , we can construct a recursive k -colorable graph G whose k -colorings code up the set of separating sets of sets A and B . Here we say C is a separating set for A and B if $A \subseteq C$ and $C \cap B = \emptyset$. Then since there is a pair of inseparable r.e. sets A and B , i.e., there is no recursive separating set for A and B , there is a recursive k -colorable graph with no recursive k -coloring. See the paper by Cenzer and Remmel [13] on Π_1^0 -classes for many examples of this type of coding.

In practice, the actual algebraic coding process may be quite easy, as in a construction of an r.e. Abelian group with an unsolvable word problem (see [7]), the old construction of a recursive field without a splitting algorithm (see [36]), or the construction of Pour-El and Richards in [74]. The coding process can also be extremely elaborate and difficult, as in Feiner's technique [34] for constructing r.e. Boolean algebras which are not isomorphic to any recursive Boolean algebra, the famous Higman embedding theorem [44], or Matijasevic's work [60, 59]. In either case, the key distinguishing feature of a coding result is the separation of the recursion theory and the classical mathematics. One performs the recursion theory in ω , gets a suitable r.e. set or pair of r.e. sets, and *then* encodes.

We remark that coding was essentially the only technique used in studying the effective content of mathematics, and in particular recursive algebra, until the papers of Metakides and Nerode [62, 63] and those of the recursive model theorists in the early 1970's. (It would seem that this is one reason for the divergence between "pure" recursion theory (on ω) and applied recursion theory). However in [62, 63] Metakides and Nerode suggested a new approach to studying the limits of the algorithmic content of mathematics. This approach did not rely on coding. It relies upon performing the construction *in* the algebraic setting, modifying all the tools of the modern recursion theory so that they specifically fit the setting at hand.

In many situations, their techniques were so successful that coding techniques have been ignored or forgotten. Recently however, within the framework developed by Metakides and Nerode, Remmel, and others, coding techniques have given some strong results (e.g., [12, 24]). Also many coding techniques give much simpler and shorter proofs of results already known, but achieved by other techniques.

Our purpose here is to systematically analyze the extent to which natural coding techniques may be utilized in the framework of recursive algebra. Our basic approach is to code a subset A of ω with which satisfies a certain property P into a distinguished “independent set” which occurs in a wide class of models. We then ask what effect this property P has on the substructure generated by A . This idea first appeared in effective Steinitz closure systems studied by Metakides and Nerode [65], Remmel and Nerode [69, 71], and Downey [19]. Remmel [82] subsequently developed a significantly more general setting than Steinitz closure systems which he called effective closure systems and which covered a much larger class of models. For the most part, we shall work in Remmel’s setting of effective closure systems in this paper.

The generality of our results requires some fairly technical definitions, but this is the basic idea. We review Remmel’s setting in Section 1.

Most of our coding results are found in Section 2. Thus we consider the effect on lattices of r.e. substructures of an “effective closure system” of encoding, for example, simple, h-simple, maximal, cohesive and other types of sets. The results of Sections 3 and 4 tend to indicate the limits of general coding results of the type of Section 2.

In Section 3, we analyze Sacks’ splitting. This is an example of a result which is true for a wide class of models but which can’t always be lifted by a coding technique.

All of our results require that the model under consideration satisfy certain structural hypotheses. The purpose of Section 4 is to illustrate the situation when these structural hypotheses do not apply. We shall see that even in quite a natural and well-behaved situation of affine subspaces of an infinite dimensional vector space over the rationals, $(V_\infty, K\ell)$, most of the features common to our other models simply don’t occur.

In Section 5 we give a few open questions.

One nice aspect of this work is that we have found many new and easy proofs of various results in the literature. For example we get easy proofs of much of [82] (whereas in [82] they were established by some rather difficult direct priority arguments).

Since our results subsume many coding results in the literature, and further we have some new easy proofs of some old results, we have felt justified in making the paper slightly survey-like. Hence we provide a large bibliography and we shall provide all the definitions so as to make the paper self contained. We hope this aids the reader.

1 Preliminaries

1.1 Basics

Let $\{\varphi_i : i \in \omega\}$ be an acceptable numbering of the partial recursive functions and let $\varphi_{e,s}(x)$ be the result (if any) after performing at most s steps in the computation of $\varphi_e(x)$. If $\varphi_{e,s}(x)$ is defined, we write $\varphi_{e,s}(x) \downarrow$. We let $W_{e,s} = \{x : x \leq s \text{ and } \varphi_{e,s}(x) \downarrow\}$. The term $\{e\}_s^A(x)$ will denote the outcome of the e -th Turing machine with oracle A after s steps on inputting x . The use function $u(A; e, x, s)$ is one plus the maximum number used in the computation if $\{e\}_s^A(x) \downarrow$, and zero otherwise. We note that since the computation of $\{e\}_s^A(x)$ takes s or fewer steps, $u(A; e, x, s) < s$. We write $A \leq_T B$ if A is Turing reducible to B and $A \equiv_T B$ if $A \leq_T B$ and $B \leq_T A$. We also write $A \not\leq_T B$ if neither $A \leq_T B$ nor $B \leq_T A$. We say $A \leq_1 B$ if there exists a one-to-one recursive function f such that $x \in A$ if and only if $f(x) \in B$. We say $A \equiv_1 B$ if $A \leq_1 B$ and $B \leq_1 A$.

Our basic setting is effective closure systems as introduced by Remmel [82]. An effective closure system $\mathcal{M} = (M, \text{cl})$ consists of a recursive set M of the natural numbers N together with an operation $\text{cl} : \mathcal{P}(M) \rightarrow \mathcal{P}(M)$, where $\mathcal{P}(M)$ denotes the power set of M , which satisfies the following:

- (i) $A \subseteq \text{cl}(A)$,
- (ii) $A \subseteq B$ implies $\text{cl}(A) \subseteq \text{cl}(B)$,
- (iii) $\text{cl}(\text{cl}(A)) = \text{cl}(A)$, and
- (iv) $x \in \text{cl}(A)$ implies that for some finite $A' \subseteq A$, $x \in \text{cl}(A')$.

Furthermore we require that cl is effective on (indices of) finite sets, i.e., we assume that there is an effective algorithm which, given $x, y_1, \dots, y_n \in M$, will decide whether or not $x \in \text{cl}(y_1, \dots, y_n)$, where $\text{cl}(y_1, \dots, y_n)$ denotes $\text{cl}(\{y, \dots, y_n\})$. We also assume that $(\mathcal{M}, \text{cl})$ always satisfy the nontriviality axiom (v) below.

- (v) $\text{cl}(\emptyset) \neq^* M$.

Here we write $A =^* B$ if there exists a finite sets, E and F , such that $\text{cl}(A \cup E) = \text{cl}(B \cup F)$. Similarly we write that $A \subseteq^* B$ if there is a finite set F such that $B \subseteq \text{cl}(A \cup F)$.

We say V is a *substructure* of $\mathcal{M}$ or V is *closed* if $V \subseteq M$ and $\text{cl}(V) = V$. It is easy to see that both the set of r.e. substructures and the set of all substructures of $\mathcal{M}$ form a lattice where the meet operation is just the set theoretic intersection and the join of two substructures V and W , denoted $V + W$, is given by $V + W = \text{cl}(V \cup W)$. We let $L(\mathcal{M})$ denote the lattice of r.e. substructures of $\mathcal{M} = (M, \text{cl})$ and $S(\mathcal{M})$ the lattice of substructures of $\mathcal{M}$.

Although we delay discussing examples until the end of this section, we pause to mention two *classes* of examples which are relevant. If $\mathcal{M}$ also satisfies

(vi) (exchange) $x \in \text{cl}(A \cup \{y\}) - \text{cl}(A)$ implies $y \in \text{cl}(A \cup \{x\})$,

we say $\mathcal{M}$ is an *effective Steinitz system*. Effective Steinitz systems have been extensively studied (e.g., [3, 4, 19, 20, 69, 71]) and provide a host of natural examples where coding is possible. Another natural class of examples are *effective algebras*. These are obtained as follows. Let (M, R) be an effective universal algebra in the sense that M is a recursive set and R a recursive set of uniformly recursive operations on M . Then we naturally associate an effective closure system (M, cl_R) with (M, R) by setting $\text{cl}_R(A)$ to be the closure of A under the operations of R and their projections. We call an effective closure system $\mathcal{M}$ formed in this way an *effective algebra*. As we shall see most natural examples such as groups, rings, fields, vector spaces, etc., are effective algebras. We remark that not all effective closure systems are effective algebras. For example, for any effective closure system $\mathcal{M} = (M, \text{cl})$, we can define an *intersection subsystem* (A, cl_A^*) for $A \subseteq M$ where for any $B \subseteq A$,

$$\text{cl}_A^*(B) = \text{cl}(B) \cap A.$$

It is easy to check that (A, cl_A^*) is an effective closure system, but not necessarily an effective algebra. Finally an effective algebra that is also a Steinitz system is called an *effective Steinitz algebra* (see [29]).

1.2 Independent Sets

For Steinitz systems such as vector spaces, the natural technique of coding is to use subsets of a recursive basis. In situations without exchange, as in Remmel [82], we shall distinguish certain “independent” subsets and code via these.

Let $\mathcal{M} = (M, \text{cl})$ be an effective system. Let $V \subseteq W$ be a pair of substructures of $\mathcal{M}$. We say that a sequence $B = \langle x_0, x_1, \dots \rangle$ is a *weak sequence* (w-sequence) for W over V if

- (1) $W = \text{cl}(V \cup B)$, and
- (2) $x_0 \notin V$ and for all $s \geq 0$, $x_{s+1} \notin \text{cl}(V \cup \{x_0, \dots, x_s\})$.

We note that if $W \not\subseteq^* V$, then every w-sequence for W over V will be infinite.

Let $D \in L(\mathcal{M})$. We say D is *decidable* if there exists an algorithm to decide whether or not $x \in \text{cl}_D(\{y_1, \dots, y_n\})$ uniformly, where

$$\text{cl}_D(A) = \text{cl}(D \cup A).$$

Now given a decidable $D \in L(\mathcal{M})$ we can define a recursive w-sequence for M over D by induction as follows. First, we let $x_0 = \mu x \in M (x \notin D)$. Then having defined $x_0, \dots, x_s$, we let

$$x_{s+1} = \mu x \in M (x \notin \text{cl}_D(x_0, \dots, x_s)).$$

Define $B = \langle x_0, x_1, \dots \rangle$.

In many cases, w-sequences do not have enough structure to be useful for our coding purposes. However, Remmel [82] introduced the notion of special sets which can be viewed as an analogue of a basis for a subspace of a vector spaces and which will be our main tool for coding.

Definition 1.1 Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and let D be a decidable substructure of $\mathcal{M}$ and V be an r.e. substructure of M which contains D . An r.e. set $S \subseteq M$ is a *special set* (*s-set*) for V over D if

- (i) for all $B \subseteq S$, $\text{cl}(D \cup B) \cap S = B$ (in particular $D \cap S = \emptyset$),
- (ii) $\text{cl}(D \cup S) = V$,
- (iii) for all $B_1, B_2 \subseteq S$, $\text{cl}(D \cup B_1) \cap \text{cl}(D \cup B_2) = \text{cl}(D \cup (B_1 \cap B_2))$.

We note that definition 1.1 is actually slightly more general than given by Remmel in [82] since Remmel only defined a special sequence for M over D . Special sets have a number of nice properties which we enumerate in our next lemma.

Lemma 1.2 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, D be a decidable substructure of $\mathcal{M}$, V be an r.e. substructure of $\mathcal{M}$, and S be a special set for V over D .*

- (a) *Given $z \in V$, there is a unique smallest finite subset F of S such that $z \in \text{cl}_D(F)$. (We refer to F as the S -support for z over D and denote it by $\text{supp}_S(z)$ (over D).)*
- (b) *$S \leq_T V$ so that if $V = M$, then S is recursive.*
- (c) *Let $s_0, s_1, \dots$ be any r.e. enumeration of S , then $\langle s_0, s_1, \dots \rangle$ is a w -sequence for V over D . In fact, any w -sequence for V over D , which satisfies part (a), induces a special set for V .*

Proof.

(a) By property (iv) of our definition of a closure system, we know that for any $x \in V$, there are finite sets $D' \subseteq D$ and $B' \subseteq S$ such that $x \in \text{cl}(D' \cup B')$. It is then easy to see that condition (iii) of our definition of an s -set implies that $F = \cap \{B \subseteq B' : x \in \text{cl}(D \cup B)\}$ is the smallest subset of S such that $x \in \text{cl}(D \cup F)$.

(b) Let $s_0, s_1, \dots$ be any r.e. enumeration of S . Note that for any n , $s_n \in \text{cl}(D \cup \{s_n\})$. Now if $s_n \in \text{cl}(D \cup \{s_0, \dots, s_{n-1}\})$, then by property (iii) of the definition of a special set, $s_n \in \text{cl}(D \cup (\{s_n\} \cap \{s_0, \dots, s_{n-1}\})) = \text{cl}(D)$, which violates the assumption that $S \cap D = \emptyset$. It follows that $\langle s_0, s_1, \dots \rangle$ is a w -sequence. It also the case that for any $x \in M$, if $x \notin V$, then $x \notin S$; and if $x \in V$, then we can find an n such that $x \in \text{cl}(D \cup \{s_0, \dots, s_n\})$, in which case $x \in S$ if and only if $x \in \{s_0, \dots, s_n\}$. Thus $S \leq_T V$.

(c) We proved in part (b) that $s_0, s_1, \dots$ is a w -sequence of S . Now suppose that for any $x \in V$, there is a smallest finite set $F = \text{supp}_S(x)$ such that $F \subseteq S$ and $x \in \text{cl}(D \cup F)$. Hence, for any sets $B_1, B_2 \subseteq S$, if $x \in \text{cl}(D \cup B_1) \cap \text{cl}(D \cup B_2)$, then $\text{supp}_S(x) \subseteq B_1$ and $\text{supp}_S(x) \subseteq B_2$. Hence $\text{supp}_S(x) \subseteq (B_1 \cap B_2)$, and thus $x \in \text{cl}(D \cup \text{supp}_S(x)) \subseteq \text{cl}(D \cup (B_1 \cap B_2))$. It follows that

$$\text{cl}(D \cup B_1) \cap \text{cl}(D \cup B_2) \subseteq \text{cl}(D \cup (B_1 \cap B_2)).$$

However, by condition (ii) of the definition of an effective closure system, $\text{cl}(D \cup (B_1 \cap B_2)) \subseteq \text{cl}(D \cup B_i)$ for $i = 1, 2$. Thus

$$\text{cl}(D \cup (B_1 \cap B_2)) \subseteq \text{cl}(D \cup B_1) \cap \text{cl}(D \cup B_2). \quad \square$$

In addition, many degree theoretic properties lift from subsets of special sets to the substructure they generate. For example, Remmel [82] proved the following.

Lemma 1.3 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, D be a decidable substructure of $\mathcal{M}$, and S be a special set for a recursive substructure V of $\mathcal{M}$ such that $V \neq^* D$. Then if W_1 and W_2 are contained in S ,*

- (i) $\text{cl}(D \cup W_1) \equiv_T W_1$,
- (ii) if $S - W_1$ is infinite, then $\text{cl}(D \cup W_1) \neq^* V$, and
- (iii) if $W_1 \subseteq W_2$ and $W_2 - W_1$ is infinite, then $\text{cl}(D \cup W_1) \neq^* \text{cl}(D \cup W_2)$.

For some results we shall require only the existence of s -sets, for others we need s -sets which satisfy some additional hypotheses. We next give two hypotheses which will be used repeatedly in what follows.

Definition 1.4

- (a) Let $X = \{x_0, x_1, \dots\}$ be an s -set for a substructure V over a decidable substructure D of an effective closure system $\mathcal{M} = (M, \text{cl})$. We say that X has the *weak exchange property* (WEP) (over D) if, given any set $Y \subseteq V$ with $\text{cl}_D(Y) \neq^* \text{cl}_D(\emptyset)$, there exists $Y' \subseteq Y$ such that
 - (i) $\text{cl}_D(Y') \neq^* \text{cl}_D(\emptyset)$, and
 - (ii) For all $y \in Y'$, y locally exchanges over X , that is, if $x_j \in \text{supp}_X(y)$ then $x_j \in \text{cl}_D((\text{supp}_X(y) - \{x_j\}) \cup \{y\})$.
- (b) We say that X has the *local exchange property* (LEP) (over D) if given any $y \in V$, y locally exchanges over X .

Examples 1.5

Sets. Let $\mathcal{M} = (\omega, \text{cl})$ where $\text{cl}(A) = A$. In this case, $L(\mathcal{M})$ is the lattice of r.e. sets. It is easy to see that ω is a special sequence.

Vector Spaces. Let V_∞ denote a fully effective infinite dimensional vector space over a computable field. That is, V_∞ consists of a recursive subset U of ω , and recursive operations for addition and scalar multiplication

on V_∞ . Moreover, we assume that V_∞ has an effective dependence algorithm, that is, that there is a uniform algorithm which given any $x, y_1, \dots, y_n$ in U , decides whether or not

$$x \in (\{y_1, \dots, y_n\})^*,$$

where $(A)^*$ denotes the subspace generated by A . In this case, $\text{cl}(A) = (A)^*$. Obviously any recursive basis of V_∞ is special. Furthermore, V_∞ is a Steinitz algebra and has (global) exchange.

Fields. Here F_∞ denotes a fully effective algebraically closed field with infinite recursive transcendence base. Here $\text{cl}(A)$ denotes the algebraic closure of A . Any recursive transcendence basis is special over $\text{cl}(\emptyset)$, the base field. Again this is a Steinitz algebra (see [65]).

Affine Spaces. In this case $\mathcal{M} = (V_\infty, K\ell)$ where V_∞ is a recursive vector space over a recursive ordered field. Define $y \in K\ell(y_1, \dots, y_n)$ if and only if $y = \sum \lambda_i y_i$ with $\sum \lambda_i = 1$. Again this is a Steinitz algebra. We denote its lattice of r.e. affine subspaces by $L(V_\infty, K\ell)$ to distinguish it from $L(V_\infty)$ (see Section 4, and see also [20]).

Locally Computable Rings and Modules. There are many other computable rings and modules which are effective closure systems. For example, consider $G = \oplus_{i \in \omega} \mathbb{Z}$, the free Abelian group on ω generators. The standard basis for G as a $\mathbb{Z}$ -module is a special sequence.

Subalgebras of Boolean Algebras. (Remmel [82]) A recursive Boolean algebra

$$\mathcal{B} = (B, \vee_{\mathcal{B}}, \wedge_{\mathcal{B}}, \neg_{\mathcal{B}})$$

consists of a recursive subset B of ω and recursive operations for the meet, $\wedge_{\mathcal{B}}$, join, $\vee_{\mathcal{B}}$, and complement, $\neg_{\mathcal{B}}$ operations which turn B into a Boolean algebra. In this case, $\text{cl}(A)$ is the subalgebra generated by A . Let $0_{\mathcal{B}}$ and $1_{\mathcal{B}}$ denote the zero and one of $\mathcal{B}$ respectively. There is a natural partial order on $\mathcal{B}$ defined by $x \leq_{\mathcal{B}} y$ if and only if $x \wedge_{\mathcal{B}} y = x$. We say an $x \in B$ is an *atom* of $\mathcal{B}$, if $x \neq 0_{\mathcal{B}}$ and there is no $y \in B$ with $0_{\mathcal{B}} <_{\mathcal{B}} y <_{\mathcal{B}} x$. We say x is an *atomless element* if $x \neq 0_{\mathcal{B}}$ and there is no atom y of $\mathcal{B}$ such that $y \leq_{\mathcal{B}} x$. Given a subalgebra C of $\mathcal{B}$, we let $\text{At}(C)$ denote the set of atoms of C . (Note that an $x \in \text{At}(C)$ is not necessarily an atom of $\mathcal{B}$.)

Remmel [79] distinguished three specific recursive Boolean algebras, namely,

- (A) $\tilde{N}$ is a recursive presentation of the Boolean algebra of finite and co-finite subset of ω , where $At(\tilde{N})$ is recursive. It is shown in [79] that any two recursive Boolean algebras which are isomorphic to $\tilde{N}$, and whose set of atoms form a recursive set, are recursively isomorphic.
- (B) $\tilde{Q}$ is a recursive presentation of the countable atomless Boolean algebra. Cantor's proof that any two countable atomless Boolean algebras are isomorphic is effective, so that $\tilde{Q}$ is unique up to recursive isomorphism. We shall think of $\tilde{Q}$ as the Boolean algebra generated by the left-closed right-open intervals of the rationals $\mathbb{Q}$.
- (C) $\tilde{C}$ is a recursive presentation of the Boolean algebra generated by the closed intervals of the rationals $\mathbb{Q}$, such that $At(\tilde{C})$ and the ideal generated by $At(\tilde{C})$ are recursive. It is shown in [79] that any two recursive Boolean algebras, which are isomorphic to $\tilde{C}$ and for which both the set of atoms and the ideal generated by the set of atoms are recursive, are recursively isomorphic.

The importance of these three recursive Boolean algebras is that Remmel [79] proved that for any given recursive Boolean algebra $\mathcal{B}$, there is a recursive Boolean algebra either of the form $\tilde{N} \times \tilde{A}$, or $\tilde{Q} \times \tilde{A}$ or $\tilde{C}$, where $\tilde{A}$ is a recursive Boolean algebra which is in the same isomorphism type as $\mathcal{B}$. We then have the following special sequences S with LEP over D in such Boolean algebras.

- (a) If $\mathcal{B} = \tilde{N} \times \tilde{A}$, then $S = \{\langle a, 0_{\tilde{D}} \rangle : a \text{ is an atom of } \tilde{N}\}$ and $D = \{\langle 0_{\tilde{N}}, a \rangle, \langle 1_{\tilde{N}}, a \rangle : a \in A\}$.
- (b) If $\mathcal{B} = \tilde{Q} \times \tilde{A}$, then $S = \{\langle [i, i+1), 0_{\tilde{A}} \rangle : i \in \omega\}$ and $D = \text{cl}(\langle x, a \rangle : x \in I \text{ and } a \in \tilde{A})$ where I is the ideal in $\tilde{Q}$ generated by those x of the form $x \subseteq (-\infty, 0)$ or for some integer $i \geq 0$, $x \subset [i, i+1)$ where $x = [j_0, j_1) \cup \dots \cup [j_{2n}, j_{2n+1})$ with $i \leq j_0 < j_1 < \dots < j_{2n+1} < i+1$.
- (c) If $\mathcal{B} = \tilde{C}$, we let $S = \{\langle [i, i+1) : i \in \omega\}$ and $D = \text{cl}(I \cup At(\tilde{C}))$ where I is the ideal of $\tilde{Q}$ described in (b) above. We refer the reader to [79, 80, 82] for further details.

For some examples we need to restrict the domain. That is, for many natural closure systems $\mathcal{M} = (M, \text{cl})$, it is the case that $\text{cl}(M) =^* \text{cl}(\emptyset)$. This may be overcome for some algebras as follows.

Ideals. For simplicity we deal with ideals in Boolean algebras $\mathcal{B}$ of the form $\tilde{N} \times \tilde{A}$, or $\tilde{Q} \times \tilde{A}$ or $\tilde{C}$, as described in the case of subalgebras of Boolean algebras above. In this case the closure operation is defined by letting $\text{cl}(A)$ equal the ideal generated by A . The problem with this case is that $\text{cl}(\{1_{\mathcal{B}}\})$ is always equal to $\mathcal{B}$. The idea is to restrict oneself to working within an appropriate maximal ideal.

$\mathcal{B} = \tilde{N} \times \tilde{A}$. Here we let our closure system be (I, cl) , where I is the ideal generated by $\langle 0_{\tilde{N}}, 1_{\tilde{A}} \rangle$ plus the set of all $\langle a, 0_{\tilde{A}} \rangle$ such that A is an atom of $\tilde{N}$. It is then easy to see that

$$S = \{ \langle a, 0_{\tilde{A}} \rangle : a \in \text{At}(\tilde{N}) \}$$

is a special set over $D = \text{cl}(\langle 0_{\tilde{N}}, 1_{\tilde{A}} \rangle)$ with LEP.

$\mathcal{B} = \tilde{Q} \times \tilde{A}$. Here we let our closure system be (J, cl) , where J is the ideal generated by $\langle 0_{\tilde{Q}}, 1_{\tilde{A}} \rangle$ plus the set of all $\langle a, 0_{\tilde{A}} \rangle$ such that $a \subseteq (-\infty, n)$ for some finite integer n . In this case, one can show that $S = \{ \langle [i, i+1), 0_{\tilde{A}} \rangle : i \in \omega \}$ is a special set with LEP over $D = \text{cl}(\langle x, 1_{\tilde{A}} \rangle)$ such that $x \subseteq (-\infty, 0)$ or $x \subset [i, i+1)$, where

$$x = [j_0, j_1) \cup \cdots \cup [j_{2n}, j_{2n+1})$$

with $i \leq j_0 < j_1 < \cdots < j_{2n+1} < i+1$. The key to proving that S is a special set with LEP is to observe that $\tilde{Q}/D$ is isomorphic to $\tilde{N}$.

$\mathcal{B} = \tilde{C}$. Here we let our closure system be (K, cl) where K is the ideal generated by the set of all a such that $a \subseteq (-\infty, n)$ for some finite integer n . In this case, one can show that $S = \{ [i, i+1) : i \in \omega \}$ is a special set with LEP over $D = \text{cl}(T)$, where T consists of the set of all x such that $x \subseteq (-\infty, 0)$ or $x \subset [i, i+1)$, where

$$x = ([j_0, j_1) \cup \cdots \cup [j_{2n}, j_{2n+1}) \vee_{\tilde{C}} F) -_{\tilde{C}} E$$

with $i \leq j_0 < j_1 < \cdots < j_{2n+1} < i+1$, and F and E are finite sets of atoms of $\tilde{C}$. The key to proving that S is a special set with LEP is to observe that $\tilde{C}/D$ is isomorphic to $\tilde{N}$.

A similar approach will work with filters of $\mathcal{B}$. See [82] for details.

Free groups. Similar problems occur in free groups. We may consider, for example, free subgroups of the free group on two generators $G = \langle x, y \rangle$. Again we simply fix a recursive subgroup G' of G which is generated by an infinite recursive independent set. Then define $(G', \text{cl}_{G'})$ in the obvious way. Then special sets (over $\emptyset$) do exist and are classical independent sets. We remark that one needs Neilson-Schreier theory for determining of independence. Also, such special sequences have LEP, see [56] for details.

Not all special sets have LEP over D , and moreover not all systems have special sequences. For example, there are many closure systems on the structure $\mathbb{Q}^n$ (the n -fold product of the rationals) given in Remmel [82].

The product ordering of the rationals ($\mathbb{Q}^n, \leq$). Then let

$$D = \{\langle 0, y_2, \dots, y_n \rangle : y_i \in \mathbb{Q}\}, \quad \text{and}$$

$$B = \{\langle x, 0, \dots, 0 \rangle : x \in \mathbb{Q} - \{0\}\},$$

then B is special with LEP over D .

The multiplicative group of the positive rationals, $\mathbb{Q}^+$. Here $\mathbb{Q}^+$ denotes the multiplicative group of the positive rationals and $\text{cl}(A)$ is the subgroup generated by A . Let $D = \text{cl}(\emptyset) = \{1\}$ and $B = \{p : p \text{ is prime}\}$. Then B is a special set over D . Note that B fails to have LEP, since for any prime p , $\text{supp}_{B_1}(p^2) = \{p\}$ but $p \notin \text{cl}(\{p^2\})$. However, if $D_1 = \text{cl}(\{p^2 : p \text{ is prime}\})$, then B is special for M over D_1 and has LEP.

Convex sets, $K(V_\infty)$. Finally, consider the structure $K(V_\infty) = (V_\infty, \langle \rangle)$ from Kalantari [46] and Downey [22]. Here we consider V_∞ where the underlying field is the rationals, $\mathbb{Q}$, and $\langle \rangle$ is the operation of taking the convex hull, viz.,

$$\langle \{x_1, \dots, x_n\} \rangle = \{y \mid y = \sum \lambda_i x_i \text{ with } \sum \lambda_i = 1 \text{ and } 0 \leq \lambda_i \leq 1\}.$$

Then $(V_\infty, \langle \rangle)$ is obviously an effective closure system.

However there are no B and D with B special over $\emptyset$ and $\text{cl}(B) = V_\infty$, as we shall prove in Theorem 4.3. However, there are w -sequences. Let

$B = \{b_0, b_1, \dots\}$ be a recursive basis of V_∞ , i.e., $(B_1)^* = V_\infty$. Define C as follows: Let $C_0 = B$. At stage $s + 1$ let

$$C_{s+1} = C_s \cup \{(s+1)b_i, -(s+1)b_j\}$$

if $(s+1) = [i, j]$ where $[,] : N \times N \rightarrow N$ is an effective pairing function. Then with the obvious ordering, $C = \bigcup_s C_s$ forms a w -sequence over $\text{cl}(\emptyset)$.

2 Coding Results

2.1 Immunity and simplicity

A set $A \subseteq \omega$ is *immune* if A is infinite and A contains no r.e. sets. A set $A \subseteq \omega$ is *hyperimmune* if A is infinite, and there is no recursive function f such that for all x and y , $D_{f(x)} \cap A \neq \emptyset$ and $x \neq y \Rightarrow D_{f(x)} \cap D_{f(y)} = \emptyset$. Here $D_0 = \emptyset$, and $D_x = \{a_0, \dots, a_n\}$ if $x > 0$ and $x = 2^{a_0} + \dots + 2^{a_n}$ where $a_0 < \dots < a_n$. An r.e. set S is *simple* (*hypersimple*) if $\omega - S$ is immune (hyperimmune). In general, given any infinite recursive set $R = \{r_0 < r_1 < \dots\}$, we say that $S \subseteq R$ is an immune (hyperimmune, simple, etc.) subset of R if the $\{i : r_i \in S\}$ is an immune (hyperimmune, simple, etc.) subset of ω .

Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system. Recall that given any two closed sets A, B of $\mathcal{M}$, we write $A =^* B$, if there are finite sets E and F contained in M such that $\text{cl}(A \cup E) = \text{cl}(B \cup F)$. With the assumption that $\text{cl}(\emptyset) \neq^* M$, we can naturally generalize many properties of the lattice of r.e. sets, $L(\omega)$. We begin with immunity. We say $V \in S(\mathcal{M})$ is *immune* if $V \neq^* \text{cl}(\emptyset)$ and for all $W \in L(M)$, if $W \subseteq V$, then $W =^* \text{cl}(\emptyset)$.

Theorem 2.1 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and let W be an r.e. substructure of $\mathcal{M}$ such that $W \neq^* \text{cl}(\emptyset)$. Let Q be an immune subset of a special set B for W over $\text{cl}(\emptyset)$. Then $\text{cl}(Q)$ is immune.*

Proof. Suppose $V \subseteq \text{cl}(Q)$ and $V \neq^* \text{cl}(\emptyset)$. As $V \subseteq \text{cl}(Q)$ for each $v \in V$, we can compute $\text{supp}_B(v)$. Moreover, for all $v \in V$, $\text{supp}_B(v) \subseteq Q$ since $V \subseteq \text{cl}(Q)$. Let $R = \bigcup \{\text{supp}_B(v) : v \in V\}$. Then R is infinite, r.e., and $R \subseteq Q$; a contradiction. $\square$

We say that S is an *effectively immune* subset of B if, given any r.e. set W contained in B , we can effectively find a finite set $F \subseteq B$ such that $W \subseteq F$.

In our setting, we say that a substructure I of $\mathcal{M}$ is *effectively immune* if given any r.e. substructure W of $\mathcal{M}$ contained in I , we can effectively compute a finite subset F contained in I such that $W \subseteq \text{cl}(F)$. Then it is easy to see that our proof of Theorem 2.1 as proves the following.

Corollary 2.2 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and let W be an r.e. substructure of $\mathcal{M}$ such that $W \neq^* \text{cl}(\emptyset)$. Let Q be an effectively immune subset of a special set B for W over $\text{cl}(\emptyset)$ with LEP. Then $\text{cl}(Q)$ is effectively immune.*

As a final corollary, if in addition W is recursive, we get co-r.e. and degree theoretic results by Lemma 2.3. Thus a (co-r.e.) immune subset of B (of degree δ) generates a (co-r.e.) immune substructure (of degree δ).

We now turn to the dual property of simplicity. In the same vein, we say that $V \in L(\mathcal{M})$ is *simple* if $V \neq^* M$ and, for all $W \in L(\mathcal{M})$, if $W \cap V = \text{cl}(\emptyset)$ then $W =^* \text{cl}(\emptyset)$. Unfortunately F_∞ (for example) contains no simple subfields. Thus we need additional axioms to guarantee the existence of simple r.e. substructures. The following condition is due to Remmel [82].

Condition 2.3 Let B be an s -set for M over D , where D is a decidable substructure with $D \neq^* M$. Then for any $R \subseteq B$ such that $B - R$ is infinite, and any $V \in L(\mathcal{M})$ such that $\text{cl}(\emptyset) \neq^* V$ and $V \cap \text{cl}_D(R) = \text{cl}(\emptyset)$, it is the case that for any finite $E \subseteq B$, there exists $v \in V$ with $\text{supp}_B(v) \cap E = \emptyset$.

The following is essentially due to Remmel [82] which we prove here only for the sake of completeness.

Theorem 2.4 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system which satisfies Condition 2.3. Let D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$, and let B is a recursive s -set for M over D . Then if Q is a hypersimple r.e. subset of B , $\text{cl}_D(Q)$ is simple.*

Proof. Let $V \neq^* \text{cl}(\emptyset)$ be an r.e. substructure of $\mathcal{M}$ such that $V \cap \text{cl}_D(Q) = \text{cl}(\emptyset)$. Then we can define an effective sequence of pairwise disjoint finite subsets of B , $D_0, D_1, \dots$, as follows. Let $D_0 = \text{supp}_B(v_0)$, where v_0 is the least element v of V such that $\text{supp}_B(v) \neq \emptyset$. Given $D_0, \dots, D_{s-1}$, let $D_s = \text{supp}_B(v_s)$, where $v_s = \mu v (v \in V \text{ and } \text{supp}_B(v) \cap (\bigcup_{i < s} D_i) = \emptyset)$. Note that v_s must exist by Condition 2.3. Then, since $v_s \notin \text{cl}_D(Q)$, $D_s \not\subseteq Q$. Hence $\{D_i\}_{i \in \omega}$ witnesses the non-h-simplicity of Q ; contradiction. $\square$

Of course, similar results on degrees and effective simplicity similarly lift from h -simple subsets. The use of h -simplicity really is necessary, as is shown by the following example. Let S be a simple but not h -simple subset of a recursive basis of V_∞ . Let $\{D_x : x \in \omega\}$ witness the non- h -simplicity of S . Then $(S)^* \cap (\{d_x : x \in \omega\})^* = \{\vec{0}\}$, where $d_x = \sum_{d \in D_x} d$. Hence $(S)^*$ is not simple.

2.2 Maximality, cohesiveness and hyperhyper-simplicity

A set $A \subseteq \omega$ is *cohesive* if A is infinite, and for any r.e. set W either $W \cap A$ or $A - W$ is finite. A set $A \subseteq B$ is *hyperhyper-immune* (*hh-immune*) if there is no recursive function f such that for all x and y ,

- (i) $W_{f(x)}$ is finite,
- (ii) $W_{f(x)} \cap A \neq \emptyset$, and
- (iii) $x \neq y \Rightarrow W_{f(x)} \cap W_{f(y)} = \emptyset$.

A collection of disjoint finite sets $\{W_{f(x)}\}_{x \in \omega}$ as above is called a *weak array*. An r.e. set $M \subseteq \omega$ is *maximal* (*hh-simple*) if $\omega - M$ is cohesive (hh -immune). Note that an r.e. set M is maximal if and only if $\omega - M$ is infinite, and for all r.e. sets $W \supseteq M$, either $\omega - W$ or $W - M$ is finite.

We say $V \in L(\mathcal{M})$ is *maximal* if $V \neq^* M$ and, for all $W \in L(\mathcal{M})$, if $W \supseteq V$ then either $W =^* M$ or $V =^* W$. Similarly $V \in S(\mathcal{M})$ is *cohesive* if $V \neq^* \text{cl}(\emptyset)$ and, for all $W \in L(M)$, either $W \cap V =^* \text{cl}(\emptyset)$ or $V \subseteq^* W$. We shall give two proofs that maximality lifts via coding. In this section, we shall give a proof based on the ideas of Downey [20]. In the next section, we give another proof using a property of maximal sets from Madan and Robinson [55]. Both techniques are extremely useful when analyzing coding properties.

Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let B be a special set for M over D and let $b_0, b_1, \dots$ be some fixed r.e. enumeration of B . Then, following [19], we define an r.e. subset of B , $B(V)$, for any $V \in L(\mathcal{M})$ with $V \supseteq D$. We call $B(V)$ the *associate* of V .

We let $B(V) = \bigcup_s B_s(V)$, where $B_0(V), B_1(V), \dots$ is defined in stages as follows. Let $v_0, v_1, \dots$ be some r.e. enumeration of V , and let $V_s = \{v_0, \dots, v_s\}$ for all s .

STAGE 0:

Let $B_0(V) = \emptyset$ and $b_i^0 = b_i$ for all i .

STAGE $s + 1$:

Let j be the least number $\leq s$ such that

$$b_j^s \in \text{cl}_D(V_s \cup \{b_i^s : i < j\})$$

and for all $k < j$,

$$b_k^s \notin \text{cl}_D(V_s \cup \{b_i^s : i < k\}).$$

If no such j exists, let $B_{s+1}(V) = B_s(V)$ and $b_{s+1}^i = b_s^i$ for all i . Otherwise, let $B_{s+1}(V) = B_s(V) \cup \{b_j\}$ and $b_i^{s+1} = b_i^s$ for $i < j$, with $b_i^{s+1} = b_{i+1}^s$ otherwise.

Note that if $Q \subseteq B$ and $V \supseteq \text{cl}_D(Q)$, then certainly $Q \subseteq B(V)$. In fact, it is easy to prove that $B - B(V) = \{b_{j_0}, b_{j_1}, \dots\}$ where $j_0 < j_1 < \dots$ is defined by

(i) $j_0 = \mu j(b_j \notin V)$, and

(ii) for $s \geq 0$, $j_{s+1} = \mu j(j > j_s \ \& \ b_j \notin \text{cl}(V \cup \{b_{j_0}, \dots, b_{j_s}\}))$.

It follows that $\text{card}(B - B(V)) = \infty$ if and only if $V \neq^* M$. Another useful observation which is straightforward to prove is that if F is a finite subset of $B - B(V)$, then $B(\text{cl}(V \cup F)) = B(V) \cup F$.

These observations combined with certain exchange properties will be crucial for our coding of maximal structures.

Theorem 2.5 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let B be a special set for M over D and let $b_0, b_1, \dots$ be some fixed r.e. enumeration of B . Then if B has WEP over D , and Q is an r.e. maximal subset of B , then $\text{cl}_D(Q)$ is a maximal substructure of $\mathcal{M}$.*

Proof. Suppose $V \in L(\mathcal{M})$ with $\text{cl}_D(Q) \subseteq V$. Form the associate $B(V)$ of V . Then $Q \subseteq B(V) \subseteq B$. If $B - B(V)$ is finite, then $V =^* M$. Hence without loss of generality, since Q is maximal, we may assume $Q \subseteq B(V)$ and $B(V) - Q$ is finite. We shall show that this assumption implies $\text{cl}_D(Q) =^* V$, and hence that V is a maximal substructure of $\mathcal{M}$. Suppose not, i.e., $V \neq^* \text{cl}_D(Q)$. Define a sequence $Z = \{z_0, z_1, \dots\}$ inductively via

$$\begin{aligned} z_0 &= \mu z \in V(z \notin \text{cl}_D(B(V))), \quad \text{and} \\ z_{s+1} &= \mu z \in V(z \notin \text{cl}(B(V) \cup \{\text{supp}_B(z_i) : i \leq s\})). \end{aligned}$$

Clearly z_s is defined for all s . By WEP, we may refine the sequence $Z = \{z_0, z_1, \dots\}$ to an r.e. sequence $Y = \{y_0, y_1, \dots\}$, so that $\text{cl}(Y) \neq^* \text{cl}(\emptyset)$, and y_i locally exchanges over B . Next let

$$a_0 = \max(\{z \mid z \in \text{supp}_B(y_0) - Q\}).$$

Then $a_0 \in \text{cl}_D(Q \cup \{b_j : b_j < a_0\} \cup \{y_0\})$ and so $a_0 \in B(V) - Q$. Next, let i_1 be the least i such that $\max\{z \mid z \in \text{supp}_B(y - 1) - Q\} > a_0$, and let

$$a_1 = \max\{z \mid z \in \text{supp}_B(y_{i_1} - 1) - Q\} > a_0.$$

Then $a_1 \in B(V) - Q$ and $a_1 > a_0$. In this way we can generate an infinite set $\{a_1, a_2, \dots\} = A$ with $A \subseteq B(V) - Q$, which contradicts the fact that $B(V) - Q$ is finite. $\square$

Thus since there are maximal sets in every high degree, we can conclude that under the hypothesis of Theorem 2.5, there exist maximal r.e. substructures of $\mathcal{M}$ in each high r.e. degree.

Next we consider the question of whether a cohesive subset of B generates a cohesive substructure. Here the situation is substantially different. For example, in $L(V_\infty)$ we have the following.

Theorem 2.6 (Shore (unpublished), Downey [21]) *If C is a co-r.e. subset of a recursive basis of V_∞ , then $(C)^*$ is not cohesive.*

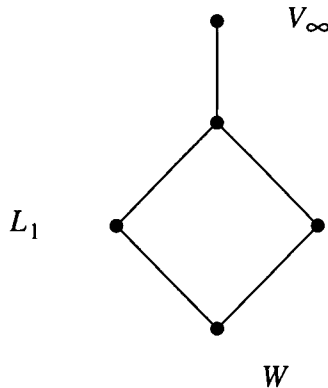
We remark that it is unclear to what extent one may generalize Theorem 2.6, even in a Steinitz closure system setting.

We now turn to *hyper-hyper-simplicity* (*hh-simplicity*). We may define a *weak array* of finitely generated nonempty closed subsets $\{V_e\}_{e \in \omega}$ of M to be one for which there is a recursive function $f(x)$, such that for all x ,

- (i) $W_{f(x)}$ is finite,
- (ii) $V_x = \text{cl}(W_{f(x)})$, and
- (iii) $\forall e (\forall x (V_e \cap \text{cl}(\bigcup_{j \neq i} V_j) = \text{cl}(\emptyset)))$.

We then define $V \in L(\mathcal{M})$ to be *hh-simple* if $V \neq^* M$ and, for every weak array of finitely generated nonempty closed subsets of M , $\{V_e\}_{e \in \omega}$, there exists an e such that $V_e \subseteq \text{cl}(V \cup (\bigcup_{j \neq e} V_j))$. Now, let $V \in L(\mathcal{M})$. We denote by $L(V, \uparrow)$ the lattice of r.e. superstructures of V , and by $L^*(V, \uparrow)$ the lattice of r.e. superstructures of V modulo the equivalence relation $=^*$. A remarkable theorem [53] of Lachlan states that in $L(\omega)$, an r.e. set A is hh-simple if and only if $L(A, \uparrow)$ is a Σ_3^0 Boolean algebra. Retzlaff [83] observed that this fails in $L(V_\infty)$. Let $L^*(V_\infty)$ denote $L(V_\infty)$ modulo finite dimensional subspaces. Lachlan's result also fails for $L^*(V_\infty)$ as the next example shows.

Example 2.7 Nerode and Smith [72] proved that every finite distributive lattice is a filter in $L^*(V_\infty)$. Thus, embed L_1 pictured below as a filter in $L^*(V_\infty)$. That is, let W be such that $L^*(W, \uparrow)$ is isomorphic to L_1 . Then W is evidently hh-simple, since $L^*(W, \uparrow)$ is finite. That is, if W is not hypersimple, then let $\{V_e\}_{e \in \omega}$ be a weak array which witnesses the non-hypersimplicity of W . Next let $U_1 \subseteq U_2$ be infinite r.e. sets such that $U_2 - U_1$ is infinite. Then let $A_i = \text{cl}(W \cup (\bigcup_{i \in U_i} V_i))$ for $i = 1, 2$. Then it is easy to show that the dimensions of both A_1/W and A_2/A_1 are infinite, and hence $A_1 \neq^* A_2$. It would then follow that $L^*(W, \uparrow)$ is infinite, which is a contradiction. Clearly $L^*(W, \uparrow)$ is not even a complemented modular lattice.



On the other hand, Retzlaff [83] observed that if $L(V, \uparrow)$ is a complemented modular lattice, then V is an hh-simple subspace. In our general setting, this is not valid, but has the same flavor as the next result which is the key to our existence theorems.

Theorem 2.8 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and let D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let B be a recursive special set for M with LEP over D , and let A be an r.e. hh-simple subset of B . Then*

(i) $L(\text{cl}_D(A), \uparrow)$ is a complemented lattice.

(ii) In fact, if $V \supseteq \text{cl}_D(A)$, there exists $Q \subseteq B$ such that

(a) $\text{cl}_D(Q)$ complements V in $L(\text{cl}_D(A), \uparrow)$, i.e.,

$$V + \text{cl}_D(Q) = M \quad \text{and} \quad V \cap \text{cl}_D(Q) = \text{cl}_D(A).$$

(b) Furthermore, if F is a finite subset of $B - B(V)$, then

$$\text{cl}_D(V \cup F) \cap \text{cl}_D(Q \cup F) = \text{cl}_D(A \cup F).$$

Proof. Let $V \supseteq \text{cl}_D(A)$. Fix an r.e. enumeration $b_0, b_1, \dots$ of B , and let $B(V)$ be the associate of V over D . Then $V + \text{cl}_D(B - B(V)) = M$, by construction. Moreover, since $A \subseteq B(V)$ and A is hh-simple, $Q = A \cup (B - B(V))$ is r.e. and $\text{cl}_D(Q) \in L(\text{cl}_D(A), \uparrow)$. Now, suppose

$$v \in (\text{cl}_D(Q) \cap V) - \text{cl}_D(A).$$

Then define $b_i = \max\{b_j : b_j \in \text{supp}_B(v) - A\}$. Notice that because $v \notin \text{cl}_D(A)$, b_i exists. Moreover, $b_i \in Q$ since $v \in \text{cl}_D(Q)$. Thus

$$v \in \text{cl}_D(A \cup \{b_j : j \leq i \text{ and } b_j \in Q\}),$$

and hence $b_i \in \text{cl}_D(A \cup \{b_j : j < i \text{ and } b_j \in Q\} \cup \{v\})$ by LEP. It follows that $b_i \in \text{cl}_D(V \cup \{b_j : j < i \text{ and } b_j \in Q\})$ since $A \cup \{v\} \in V$. However by the construction of $B(V)$, this means that $b_i \in B(V)$ which contradicts the fact that $b_i \in A \cup (B - B(V)) = Q$. Thus there can be no such v and hence $V \cap \text{cl}_D(Q) = \text{cl}_D(A)$. This proves (i) and (ii) (a).

Finally we need to prove (ii) (b). Let F be a finite subset of $B - B(V)$ and set $V' = \text{cl}_D(V \cup F)$ and $Q' = Q \cup F$. Again suppose $v \in V' \cap \text{cl}_D(Q')$ and $v \notin \text{cl}_D(A \cup F)$. Using a very similar argument, we can show that the maximum member of $Q' = (B - B(V)) \cup F$ in $\text{supp}_B(v)$ ought to be in $B(V') - F = (B(V) \cup F) - F = B(V)$, giving a contradiction. We leave the details to the reader. $\square$

Theorem 2.9 *Let B satisfy the hypotheses of Theorem 2.8. Then*

- (i) *if A is an hh-simple r.e. subset of B , then $\text{cl}_D(A)$ is hh-simple.*
- (ii) *Furthermore, if A is atomless (i.e., is not contained in any maximal r.e. subset of B), then $\text{cl}_D(A)$ is not contained in any maximal r.e. substructure of $\mathcal{M}$.*

Proof.

(i) Suppose not. Let $\{V_i\}_{i \in \omega}$ be a weak array witnessing the non-hh-simplicity of $\text{cl}_D(A)$. We first construct $Q \supset \text{cl}(A)$ with a strong “noncomplemented” property, and then appeal to Theorem 2.8 for a contradiction. The argument is along the lines of one in [83]. Specifically, let $\{Q_i\}_{i \in \omega}$ list the r.e. subsets of B . We construct $R = \bigcup_s R_s$ in stages. Let $V_n^s (Q_{n,s})$ denote the elements enumerated into $V_n (Q_n)$ after s steps. Fix some enumeration of A , $\{A^s\}_{s \in \omega}$, such that $A_{s+1} - A_s$ has cardinality 1 for all s .

STAGE 0:

Let $R_0 = \text{cl}_D(\emptyset)$.

STAGE $s > 0$:

For each $n < s$, if $a \in V_n^s \cap (\text{cl}(R_s \cup Q_{n,s}) - \text{cl}(R_s))$, put a into R_{s+1} . Also put b_j into R_{s+1} , where $b_j \in A_{s+1} - A_s$.

Set $R = \text{cl}(\bigcup_s R_s)$.

Now suppose $Q_e \subseteq B$, and that $\text{cl}_D(Q_e)$ is the complement of R in $L(\text{cl}_D(A), \uparrow)$ given by Theorem 2.8 (ii). Let $\tilde{V}_n = V_n - \text{cl}_D(A \cup (\bigcup_{m \neq n} \{V_m\}))$. As the V_i form a weak array witnessing the non-hh-simplicity of $\text{cl}_D(A)$, $\tilde{V}_n \neq \text{cl}(\emptyset)$. Also, $\tilde{V}_n \subseteq V_n \subseteq Q_e + R = M$. Next, let s be the least stage such that for some $a \in \tilde{V}_n$, $a \in \text{cl}_D(Q_{e,s} \cup R_s)$. By our choice of s and $\tilde{V}_n$, we see that $\text{cl}_D(R_s) \subseteq \text{cl}_D(A \cup (\bigcup_{m \neq n} V_m))$. Hence $a \notin \text{cl}_D(R_s)$, and so $a \in R_{s+1}$.

Now, $a \in \text{cl}_D(R_s \cup Q_{e,s})$. Find a minimal finite subset $F = \{b_1, \dots, b_n\} \subseteq Q_e - A$, which is the support of a over $\text{cl}_D(R_s)$, where $F \subseteq Q_{e,s}$. By LEP, it follows that

$$b_1 \in \text{cl}_D(A \cup \{b_2, \dots, b_n\} \cup \{a\}) \subseteq \text{cl}_D(R \cup \{b_2, \dots, b_n\}).$$

But also,

$$b_1 \in \text{cl}_D(Q_e) = \text{cl}_D(Q_e \cup \{b_2, \dots, b_n\}).$$

Hence

$\text{cl}_D(R \cup \{b_2, \dots, b_n\}) \cap \text{cl}_D(Q_e \cup \{b_2, \dots, b_n\}) \neq \text{cl}_D(A \cup \{b_2, \dots, b_n\})$
 contradicting Theorem 2.8 (ii) (b). Hence $\text{cl}_D(A)$ is hh-simple.

For (ii), let $V \in L(\mathcal{M})$ be maximal with $\text{cl}_D(A) \subseteq V$, and suppose A is hh-simple and atomless. Again let $Q = (B - B(V)) \cup A$, so that $\text{cl}_D(Q)$ is a complement of V in $L(\text{cl}_D(A), \uparrow)$. As A is atomless, there exists an r.e. set R with $B(V) \subseteq R \subseteq B$ and $\text{card}(B - R) = \text{card}(R - B(V)) = \infty$. Let $R' = R \cap Q$. Our construction of $B(V)$ and LEP ensure that $V \neq^* \text{cl}_D(R') \neq^* M$. Hence $\text{cl}_D(A)$ is atomless. $\square$

A natural question is what does $L(V, \uparrow)$ look like if V is hh-simple? In general, we don't know. However in [24], Downey discovered a property which is relevant to this question and to the results of the next section.

Definition 2.10 ([24]) We say that A has the *lifting property* if, given any strong array $\{D_{f(x)}\}_{x \in \omega}$ where f is a recursive function, $\text{card}(D_{f(x)} - A) \leq 1$ for almost all x .

In [24] it is shown that

Theorem 2.11 (Downey [24]) *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and let D be a decidable substructure of $\mathcal{M}$. Suppose K is a special set with LEP over D . Let $A \subseteq K$ be r.e. and have the lifting property. Then for any $W \in L(\mathcal{M})$ with $W \supseteq \text{cl}_D(A)$, there is an r.e. $Q \subseteq K$ such that $\text{cl}_V(Q) =^* W$. Hence in particular, $L^*(A, \uparrow)$ in $L(K)$ is recursively isomorphic to $L^*(\text{cl}_D(A), \uparrow)$ in $L(\mathcal{M})$.*

Proof. Given $W \in L(\mathcal{M})$ with $W \supseteq \text{cl}_D(A)$, we can effectively list the elements $w_0, w_1, \dots$ of W which are not in D . Then since K is a special set, it is easy to see that there is a recursive function f such that $D_{f(x)} = \text{supp}_K(w_x)$ for all x . Then since A has the lifting property, $S = \{i : \text{card}(D_{f(i)} - A) \geq 2\}$ is a finite set. Let $F = \bigcup_{i \in S} D_{f(i)}$ and $Q = A \cup (\bigcup_{i \in \omega} D_{f(i)})$. Then clearly, $\text{cl}_D(W \cup F) \subseteq \text{cl}_D(Q)$. On the other hand, if $i \notin S$, let $b_{i,j}$ be the unique element of $D_{f(i)} - A$. Then by LEP,

$$b_{i,j} \in \text{cl}_D((\text{supp}_K(w_i) - \{b_{i,j}\}) \cup \{w_i\}),$$

and hence $b_{i,j} \in \text{cl}(W \cup F)$. Thus $\text{cl}_D(Q) \subseteq \text{cl}(F \cup W)$. $\square$

Now, given any Σ_3^0 Boolean algebra A , one can modify (as in [24]) the standard argument to construct an r.e. hh-simple set B whose lattice of supersets is isomorphic to A , such that B has the lifting property. Consequences of this (from [24]), are that if $\mathcal{M}$ is as above, then $L(\mathcal{M})$ is not recursively presentable. We remark that, using the lifting property in [24], it is also shown that $L(\mathcal{M})$ is undecidable. We examine a related notion (“weakly co-1-1”) in the next section.

2.3 r -maximality and major substructures

We say that an r.e. set $A \subseteq \omega$ is r -maximal if $(\omega - A)$ is infinite and, for any recursive set R , either $A \cup R$ or $A \cup (\omega - R)$ is cofinite. If $A \subseteq B$ are r.e. sets, we say that A is a *major subset* of B if, for every r.e. set W , $(\omega - B) \subseteq^* W \Rightarrow (\omega - A) \subseteq^* W$. If $A = \{a_0 < a_1 < \dots\}$ is an infinite set, then the principal function of A , p_A , is defined by $p_A(n) = a_n$. We say that a function $f : \omega \rightarrow \omega$ *majorizes* a function $g : \omega \rightarrow \omega$ if $f(x) \geq g(x)$ for all x . We say that a function $f : \omega \rightarrow \omega$ *dominates* a function $g : \omega \rightarrow \omega$ if $f(x) \geq g(x)$ for all but finitely many x . We say that A is *dense simple* if A is r.e., $(\omega - A)$ is infinite, and p_A dominates every recursive function.

We say $V \in L(\mathcal{M})$ is r -maximal if $V \neq^* M$ and, whenever $W_1 + W_2 = M$ for $W_1, W_2 \in L(\mathcal{M})$, either $V + W_1 =^* M$ or $V + W_2 =^* M$. For $V, W \in L(\mathcal{M})$ we say V is a *major substructure* of W if $V \subseteq W$, $V \neq^* W$, and whenever $W + Q = M$ for $Q \in L(\mathcal{M})$, $V + Q =^* M$. For major substructures we have the following result essentially due to Downey [20].

Theorem 2.12 (Downey) *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and let D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let B be a recursive s -set for M over D , and let A and C be r.e. subsets of B such that A is a major subset of C . Then $\text{cl}_D(A)$ is a major substructure in $\text{cl}_D(C)$.*

Proof. Suppose $W + \text{cl}_D(C) = M$. Compute r.e. sets K, N as follows.

STAGE 0:

Set $K_0 = \emptyset = N_0$.

STAGE $s + 1$:

First compute the least $y \in C$ with $y \notin \text{cl}_D(K_s \cup N_s)$. Set $K_{s+1} = K_s \cup \{y\}$. Now compute the least $z \in W$ such that $z \notin \text{cl}_D(K_{s+1} \cup N_s)$. Set $N_{s+1} = N_s \cup \{z\}$.

Let $K = \bigcup_s K_s$ and $N = \bigcup_s N_s$. Then $K \subseteq C$, and $K \cup N$ is an r.e. w-sequence for M . Thus $K \cup N$ is recursive. Hence K is recursive. Thus $K \subseteq^* A$ (as A is a major subset of C). Hence

$$\text{cl}_D(A) + W =^* \text{cl}_D(K) + W \supseteq \text{cl}_D(K) + \text{cl}_D(N) = M. \quad \square$$

Corollary 2.13 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and let D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. There exist non-maximal r-maximal r.e. substructures in each high r.e. degree whenever M contains an infinite recursive s-set B for M with WEP over D .*

Proof. Let C be a maximal subset of B , and let A be a major subset of C . Then by Theorem 2.5, $\text{cl}_D(C)$ is a maximal substructure of $\mathcal{M}$. Suppose that W_1 and W_2 are r.e. substructures of $\mathcal{M}$ such that $W_1 + W_2 = M$. Then for $i = 1, 2$, either

$$\text{cl}_D(C) + W_i =^* \text{cl}_D(C) \quad \text{or} \quad \text{cl}_D(C) + W_i =^* M.$$

Since we cannot have $\text{cl}_D(C) + W_i =^* \text{cl}_D(C)$ for both $i = 1$ and $i = 2$, there must be some i such that $\text{cl}_D(W) + W_i =^* M$. But as $\text{cl}_D(A)$ is a major substructure of $\mathcal{M}$, $\text{cl}_D(A) + W_i =^* M$. Thus $\text{cl}_D(A)$ is r-maximal. However, $\text{cl}_D(C)$ witnesses that $\text{cl}_D(A)$ is not maximal. $\square$

We cannot, at this stage, prove (even) that an r-maximal r.e. subset of a recursive basis of V_∞ generates an r-maximal subspace. This is related to the following open question.

Question. Suppose that $W_1 \oplus W_2 = V_\infty$ and that B is a basis of V_∞ . Do there exist disjoint subsets B_1, B_2 with $B_1 \cup B_2 = B$, and such that $W_1 \oplus B_1^* = W_2 \oplus B_2^* = V_\infty$? If W_1 and W_2 are decidable, do recursive B_1 and B_2 exist?

We can, however, lift the existence of atomless r-maximal substructures from results in $L(\omega)$. For this we need some new tools similar to those developed in Downey [24].

Now Madan and Robinson [55] defined a set $A \subseteq \omega$ to be 1-1 if every recursive function is eventually constant or eventually one-to-one on A . That is, for all recursive f , there is a t such that either for all $s > t$, $f(s) = f(t)$ if $s \in A$, or for all $s_1, s_2 > t$, $s_1, s_2 \in A$ implies $f(s_1) \neq f(s_2)$. A set C is co-1-1 if $\omega - C$ is 1-1. Madan and Robinson [55] proved the following.

Theorem 2.14 (Madan and Robinson)

- (i) *Maximal r.e. sets are co-1-1.*
- (ii) *Major subsets of co-1-1 sets are co-1-1.*
- (iii) *There is an r.e. r -maximal dense simple set that is not co-1-1.*

For our purposes, we need an (apparently) weaker notion concerning only f 's of a special type. We formulate this in terms of strong arrays.

Definition 2.15 We say that an r.e. set A is *weakly co-1-1* if, given any strong array $\{D_{g(x)}\}_{x \in \omega}$ where g is a recursive one-to-one function such that $\bigcup x D_{g(x)} = \omega$, then for almost all x , $\text{card}(D_{g(x)} - A) \leq 1$.

The crucial difference between a set A having the lifting property and being weakly co-1-1 is that the condition " $\bigcup x D_x = \omega$ " is missing from the definition of the lifting property.

It is easy to see that co-1-1 implies weakly co-1-1 as follows: Let $\{D_{g(x)}\}_{x \in \omega}$ be a strong array with union ω as in the definition of weakly co-1-1. Define f via $f(y) = x$ for all y in $D_{g(x)}$. If A is co-1-1, then f is either eventually constant on $(\omega - A)$ or f is eventually one-to-one on $(\omega - A)$. The former is impossible, the latter gives the desired result.

Now let $V \in L(\mathcal{M})$, and B be an s -set for M over D . We define the *support* of V , $\text{supp}_B(V)$, to be

$$\text{supp}_B(V) = \bigcup \{\text{supp}_B(v) : v \in V\}.$$

The key lemma is:

Lemma 2.16 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let B be a recursive s -set with LEP for M over D and let A be a weakly co-1-1 subset of B . Suppose $V \supseteq A$, $V \in L(\mathcal{M})$, and $\text{supp}_B(V) =^* B$. Then $V =^* M$.*

Proof. Since $\text{supp}_B(V) =^* B$, let H_0 be the finite set $B - \text{supp}_B(V)$. Let

$$\begin{aligned} x_1 &= \mu x \in B (x \notin F), \\ v_1 &= \mu v \in V (x_1 \in \text{supp}_B(v)), \quad \text{and} \\ H_1 &= \text{supp}_B(v_1). \end{aligned}$$

By induction, let

$$\begin{aligned} x_{s+1} &= \mu x \in B(x \notin F \cup (\bigcup_{i=1}^s H_i)), \\ v_{s+1} &= \mu v \in V(x_{s+1} \in \text{supp}_B(v)), \quad \text{and} \\ H_{s+1} &= \text{supp}_B(v_{s+1}). \end{aligned}$$

It is easy to see that x_s , v_s and H_s are defined for all s , since $B = F \cup \text{supp}_B(V)$. Thus $\bigcup_x H_x = B$, and $\{H_x\}$ is a strong array. By hypothesis, we may suppose that for all $x > t$, $\text{card}(H_x - A) \leq 1$ for some t . However, we can now see by induction that $(\forall x > t)(H_x \subseteq \text{cl}(V \cup (\bigcup_{i \leq t} H_i)))$. Consider H_t . Now $H_t = \text{supp}(v_t) = \{b_0, \dots, b_n\}$, say. Then for at most one j , $b_j \notin A$. By LEP,

$$b_j \in \text{cl}((\{b_0, \dots, b_n\} - \{b_j\}) \cup \{v_i\}) \subseteq V.$$

Hence we can suppose for an induction that $\forall t < k \leq n(H_k \subseteq V)$. A similar use of LEP will show $H_{n+1} \subseteq V$, giving the result. $\square$

This allows us to give alternative proofs for some earlier results.

Corollary 2.17 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let B be a recursive s -set for M with LEP over D . Then if $A, C \subseteq B$ with A a maximal subset of B and C a major subset of A , then $\text{cl}_D(A)$ is maximal, and $\text{cl}_D(C)$ is r -maximal and is a major substructure of $\text{cl}_D(A)$.*

Proof. By Theorem 2.14 (i) and (ii), both A and C are co-1-1. Let $V \supseteq \text{cl}_D(A)$. If $\text{supp}_B(V) \neq^* B$, then since $\text{supp}_B(V) \supseteq A$, $\text{supp}_B(V) =^* A$, and hence $V =^* \text{cl}_D(A)$. Otherwise $\text{supp}_B(V) =^* B$. But then by Lemma 2.16, $V =^* M$. Hence $\text{cl}_D(A)$ is maximal. The second part is similar, and follows from the next result. $\square$

In general, we have the following.

Theorem 2.18 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let B be a recursive s -set for M with LEP over D .*

- (i) *Then if A is an r -maximal weakly co-1-1 subset of B , $\text{cl}_D(A)$ is r -maximal.*
- (ii) *Furthermore, if A is also atomless, then so is $\text{cl}_D(A)$.*

Proof.

(i) Let $W_1 + W_2 = M$ with $W_1, W_2 \in L(\mathcal{M})$, and let A, B satisfy the hypotheses of Theorem 2.18 (i).

Case 1. $\text{supp}_B(W_1 + \text{cl}_D(A)) =^* B$.

Then by Lemma 2.16, $W_1 + \text{cl}_D(A) =^* M$.

Case 2. $\text{supp}_B(W_1 + \text{cl}_D(A)) \neq^* B$.

Then certainly $\text{supp}_B(W_1) \neq^* B$. Now evidently

$$\text{supp}_B(W_1) \cup \text{supp}_B(W_2) = B.$$

Hence, as A is r -maximal, $A \cup \text{supp}_B(W_2) =^* B$. But $\text{cl}_D(A) \subseteq \text{cl}_D(A) + W_2$ and $\text{supp}_B(\text{cl}_D(A) + W_2) =^* B$, so that by Lemma 2.16, $\text{cl}_D(A) + W_2 =^* M$. Thus $\text{cl}_D(A)$ is r -maximal.

(ii) Suppose furthermore that A is atomless, but $\text{cl}_D(A)$ is not. Let $Q \in L(\mathcal{M})$ with Q maximal and $\text{cl}_D(A) \subseteq Q$. Now consider $\text{supp}_B(Q)$. Clearly $A \subseteq \text{supp}_B(Q)$. By Lemma 2.16, it must be the case that $\text{supp}_B(Q) \neq^* B$. As A is atomless, there is an r.e. set P with $\text{supp}_B(Q) \subseteq P \subseteq B$ and

$$\text{card}(B - P) = \text{card}(P - \text{supp}_B(Q)) = \infty.$$

But then $Q \subseteq \text{cl}_D(P)$, $\text{cl}_D(P) \neq^* Q$, and $M \neq^* \text{cl}_D(P)$. Thus $\text{cl}_D(Q)$ is not maximal. $\square$

Downey [24] essentially proved the following.

Theorem 2.19 (Downey) *There exist atomless weakly co-1-1 r -maximal r.e. sets. In fact, these may be constructed in each high r.e. degree, and we may construct these sets to have the lifting property.*

Proof. We note that using Martin permitting and coding in the usual way (e.g., [95]) will not affect the construction in [24]. See [24] for further details. $\square$

Corollary 2.20 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let δ be any nonzero high r.e. degree, and suppose M has a recursive s -set with LEP (over D). Then M has an atomless r -maximal substructure of degree δ .*

On the other hand, the construction of Madan and Robinson [55] actually shows (extending Theorem 2.14 (iii)):

Theorem 2.21 (Madan and Robinson) *There is an r.e. dense simple r -maximal set that is not weakly co-1-1.*

We remark that we do not know if being weakly co-1-1 is strictly weaker than possessing the lifting property. Moreover it is unknown whether major subsets of maximal sets have the lifting property. Thus it seems conceivable, in view of Theorem 2.21, that perhaps there are r -maximal subsets of recursive bases that do not generate r -maximal subspaces.

2.4 Creativity and nowhere simplicity

We now turn to effectively non-complemented but nonsimple substructures. We say that N is a *nowhere simple* subset of ω if

- (i) N is r.e.,
- (ii) $\omega - N$ is infinite, and
- (iii) for any r.e. set W_e , there is an r.e. set $W_f \subseteq W_e - N$ such that W_f is infinite if and only if $W_e - N$ is infinite.

N is said to be *effectively nowhere simple* if

- (i) there is a recursive function f such that for all e , $W_{f(e)} \subseteq W_e - N$, and
- (ii) $W_{f(e)}$ is infinite if and only if $W_e - N$ is infinite.

An r.e. set $C \subseteq \omega$ is *creative* if there is a partial recursive function h such that for all e , if $W_e \cap C = \emptyset$, then $h(e)$ is defined and $h(e) \notin W_e \cup C$.

Analogously to the $L(V_\infty)$ case from [19] and [69], we define $V \in L(\mathcal{M})$ to be a *nowhere simple substructure* if

- (i) $V \neq^* M$, and
- (ii) for all $W \in L(\mathcal{M})$, if
 - (a) there exists $W' \in L(\mathcal{M})$ with $W' \subseteq W$ and $W' \cap V = \text{cl}(\emptyset)$, then
 - (b) $W' \neq^* \text{cl}(\emptyset)$ if and only if $\text{cl}(W \cup V) \neq^* \text{cl}(V)$.

Furthermore, if we can effectively compute an index for W' from one for W , we say that V is an *effectively nowhere simple substructure*. We say that $C \in L(\mathcal{M})$ is a *creative substructure* if

- (i) $C \neq^* M$, and
- (ii) there is a partial recursive function h such that, for all e with $W_e \subseteq M$ and $\text{cl}(W_e) \cap C = \text{cl}(\emptyset)$,
 - (a) $h(e) \downarrow$,
 - (b) $h(e) \in M - (\text{cl}(W_e) \cup C)$, and
 - (c) $\text{cl}(W_e \cup \{h(e)\}) \cap C = \text{cl}(\emptyset)$.

Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and D be a decidable substructure of M such that $D \neq^* M$. To construct a nowhere simple and effectively nowhere simple substructure, it will suffice to assume that there is an infinite recursive s -set R for $\mathcal{M}$ over D which satisfies the following property.

Condition 2.22 For any disjoint infinite subsets R_1, R_2 of R , and any sequence $\{v_0, v_1, \dots\}$ such that for all i ,

$$(\text{supp}_R(v_i) \cap R_1) - (R_2 \cup (\bigcup_{j \neq i} \text{supp}(v_j))) \neq \emptyset,$$

it is the case that

$$\text{cl}(V) \cap \text{cl}_D(R_2) = \text{cl}(\emptyset) \quad \text{and} \quad \text{cl}(V) \neq^* \text{cl}(\emptyset),$$

where $V = \text{cl}(\{v_0, v_1, \dots\})$.

Theorem 2.23 Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system and D be a decidable substructure of M such that $D \neq^* M$. Suppose B is a recursive s -set for M over D , and that B satisfies Condition 2.22. Then if A is any (effectively) nowhere simple subset of B , $\text{cl}_D(A)$ is an (effectively) nowhere simple substructure of M .

Proof. We only give the proof of the effectively nowhere simple case, as the proof of the nowhere simple case is essentially the same. Let $W \in L(\mathcal{M})$ with

$$W + \text{cl}_D(A) \neq^* \text{cl}_D(A).$$

Consider $\text{supp}_B(\text{cl}_D(A) + W)$. Clearly $W + \text{cl}_D(A) \neq^* \text{cl}_D(A)$ if and only if $(\text{supp}_B(\text{cl}_D(A) + W) - A)$ is infinite. Now, as A is effectively nowhere simple, we can effectively find $Q \subseteq \text{supp}_B(\text{cl}_D(A) + W)$ such that $Q \cap A = \emptyset$, Q is r.e., and $\text{card}(Q) = \infty$, if and only if $(\text{supp}_B(\text{cl}_D(A) + W) - A)$ is infinite. Now it follows that $\text{cl}(\emptyset) \neq^* \text{cl}(Q)$ if and only if Q is infinite. Construct $\{v_0, v_1, \dots\}$ as follows:

STAGE 0:

Compute $v_0 \in W$ such that $(\text{supp}_B(v_0) \cap Q) \neq \emptyset$.
Set $R_0 = \text{supp}_B(v_0)$.

STAGE $s + 1$:

Compute $v_{s+1} \in W$ such that $\text{supp}_B(v_{s+1}) \cap Q - R_s \neq \emptyset$.
Set $R_{s+1} = R_s \cup \{\text{supp}_B(v_{s+1})\}$.

Then $\{v_0, v_1, \dots\}$ satisfies Condition 2.22 with $R_1 = Q$ and $R_2 = A$. Thus if $V = \text{cl}(\{v_0, v_1, \dots\})$, then $\text{cl}(V) \cap \text{cl}_D(A) = \text{cl}(\emptyset)$. Finally, $\text{cl}(V) \subseteq W$ by construction. $\square$

We remark that most “algebraic” systems, and in particular algebras like $\tilde{N}$, V_∞ and F_∞ , satisfy Condition 2.22. In fact, one can show that any Steinitz system satisfies Condition 2.22, which gives Nerode and Remmel’s result from [69]. Other applications include Boolean algebras and ideals. On the negative side, we cannot replace “(effectively)” by “(noneffectively)” in the statement of (2.23) since Downey and Remmel proved the following result.

Theorem 2.24 (Downey and Remmel [31]) *In each nonzero r.e. degree, there exists a noneffectively nowhere simple subset of a recursive basis of V_∞ which generates an effectively nowhere simple subspace of V_∞ .*

To construct a noneffectively nowhere simple substructure, we need to (apparently) resort to a direct construction.

Theorem 2.25 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and let D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let δ be any nonzero r.e. degree, and suppose B is a recursive special sequence for M over D which satisfies Conditions 2.3 and 2.22. Then there exists an r.e. set $A \subseteq B$ such that $\text{cl}_D(A)$ is a noneffectively nowhere simple substructure of degree δ .*

Proof. Our proof is based on the existence proof for a noneffectively nowhere simple subspace of V_∞ given in [31]. Let C be an r.e. set of degree δ , and let $b_0 < b_1 < \dots$ be the increasing enumeration of B . Let

$$R = \{b_{\langle x, i \rangle} : x \in C \text{ and } i \in \omega\},$$

where $\langle \cdot, \cdot \rangle$ is some recursive pairing function which maps $\omega \times \omega$ onto ω . For any subset E of B , let $E^{(n)} = \{b_{\langle n, i \rangle} : b_{\langle n, i \rangle} \in E\}$. Let $h(\omega) = R$ be a one-to-one recursive enumeration of R . We construct A in stages. Let $U_i = \text{cl}(W_i \cap M)$ for all i , where W_i is the i -th r.e. set. Let

$$U_{e,s} = \{x : x \leq s \text{ \& } x \in \text{cl}(W_{e,s} \cap M)\}.$$

For each e , we employ a possibly infinite set of Δ_e markers in the course of the construction which will help us witness that $\text{cl}_D(A)$ is nowhere simple.

STAGE s :

For each $e < n \leq s$, if there is $y \in U_{e,s}$ such that $b_{\langle r, i \rangle} \in \text{supp}_B(y)$ for some i and no $b_{\langle n, j \rangle}$ has a Δ_e marker on it, put a Δ_e marker on $b_{\langle n, k \rangle}$, where k is the least such i . Then enumerate $h(s)$ into A unless it has a Δ_e marker on it.

Lemma 2.26 *A is a noneffectively nowhere simple subset of B such that $A \equiv_T C$.*

Proof. Let $B_0, B_1, \dots$ be an effective list of all r.e. subsets of B . Now if A is an effectively nowhere simple subset of B , there would be a recursive function f such that

(i) $B_{f(e)} \subseteq B_e - A$, for all e , and

(ii) $\text{card}(B_{f(e)}) = \infty$ iff $\text{card}(B_e - A) = \infty$.

Now it is easy to see that $A^{(n)} \subseteq R^{(n)}$ for all n . In fact, $\text{card}(R^{(n)} - A^{(n)}) \leq n + 1$, since at most $n + 1$ elements of $R^{(n)}$ can have Δ_e markers on them, i.e., at most one for each $e \leq n$. Then let u be a recursive function such that $B^{(n)} = B_{u(n)}$ for all n . Now suppose $n \notin C$, so that $A^{(n)} = R^{(n)} = \emptyset$. Then $B_{u(n)} = (B^{(n)})^*$ is disjoint from A , so then $B_{f(u(n))}$ must be infinite. On the other hand if $n \in C$, then $R^{(n)} = B^{(n)}$, so that $\text{card}(R^{(n)} - A^{(n)}) \leq n + 1$ and hence $\text{card}(B_{f(u(n))}) \leq n + 1$. Thus $n \notin C$ if and only if $\text{card}(B_{f(u(n))}) > n + 1$,

which would imply that $B - C$ is r.e., and hence C would be recursive, contrary to our hypothesis. Thus A is not an effectively nowhere simple subset of R .

Now A is a nowhere simple subset of B . For suppose that $B_e - A$ is infinite. Then either for some n , $B_e^{(n)} - A$ is infinite, which means that $n \notin C$ and $A^{(n)} = \emptyset$, so then $B_e^{(n)}$ is an infinite r.e. subset of B contained in $B_e - A$. Otherwise, for infinitely many n , $B_e^{(n)} - A^{(n)} \neq \emptyset$, in which case, if $V_k = \text{cl}_D(B_e)$, there will be infinitely many Δ_k markers placed, and hence the set of elements with Δ_k markers on them will form an infinite r.e. subset of B contained in $B_e - A$.

Finally, note that $x \in A$ if and only if $x = h(s) \in R$, and x is put into A at stage s . Thus $A \leq_T R \equiv_T C$. By our observations above, $x \in C$ if and only if $A \cap \{b_{\langle n, 0 \rangle}, b_{\langle n, 1 \rangle}, \dots, b_{\langle n, n+1 \rangle}\} \neq \emptyset$, so that $C \leq_T A$. $\square$

Since A is a nowhere simple subset of B , it follows from Theorem 2.23 that $\text{cl}_D(A)$ is a nowhere simple substructure of $\mathcal{M}$. Thus to complete the proof of our theorem, we need only prove the following.

Lemma 2.27 *$\text{cl}_D(A)$ is not an effectively nowhere simple substructure of $\mathcal{M}$.*

Proof. Suppose that $\text{cl}_D(A)$ is an effectively nowhere simple substructure. Let $Q_n = B^{(n)}$. If $n \notin C$, then $A^{(n)} = R^{(n)} = \emptyset$.

If $n \in C$, then at most n elements of $B^{(n)}$ can be kept out of A . Now let f be a witness function for the effective simplicity of $\text{cl}_D(A)$. That is, suppose that for all e ,

- (i) $U_{f(e)} \subseteq U_e$,
- (ii) $U_{f(e)} \cap \text{cl}_D(A) = \text{cl}(\emptyset)$, and
- (iii) if $\text{cl}_D(U_e \cup A) \neq^* \text{cl}_D(A)$ then $U_{f(e)} \neq^* \text{cl}(\emptyset)$.

Let $U_{g(n)} = \text{cl}_D(Q_n)$. Notice that $e \in C$ if and only if $\text{cl}_D(U_{g(e)} \cup A) =^* \text{cl}_D(A)$. Thus by Condition 2.3, $e \notin C$ if and only if $\text{cl}(U_{f(g(e))}) \neq^* \text{cl}(\emptyset)$; but this latter holds if and only if there exist $n+1$ elements $y_1, \dots, y_{n+1}$ of $\text{cl}(U_{f(g(e))})$ such that for all i , $\text{supp}_B(y_i) \neq \emptyset$ and for all $i \neq j$, $\text{supp}_B(y_i) \cap \text{supp}_B(y_j) = \emptyset$. But this would imply that $\omega - C$ is r.e., and hence C would be recursive. Thus no such f exists. $\square$ $\square$

We note that we do not know if Condition 2.22 can be removed. We refer the reader to [31] for further details on nowhere simple subspaces of V_∞ . Our final results for this subsection concern creativity.

Theorem 2.28 *Let M be an effective Steinitz algebra, and A a creative subset of a recursive basis for M . Then $\text{cl}(A)$ is creative.*

Proof. It is routine to construct a creative subset A' of B such that $\text{cl}(A')$ is creative. By Myhill's theorem there is a recursive permutation of B taking A to A' . As $\mathcal{M}$ is a Steinitz algebra, this map lifts to a recursive automorphism of $L(\mathcal{M})$. Hence $\text{cl}(A)$ is creative. $\square$

In the general setting, again life is not so nice. The answer was given by Remmel [82], but by using a direct construction. We state the associated condition and the result for completeness.

Definition 2.29 (Remmel [82]) We say a one-to-one recursive function $f : \omega \rightarrow (M - \text{cl}(\emptyset))$ is *special over A* , where $A \in L(\mathcal{M})$, if A is recursive, and if there is an effective sequence $S_0, S_1, \dots$ of pairwise disjoint sets such that

- (1) $\forall i (f(i) \in S_i)$,
- (2) given any finite set Q , and $x \in M$, we can decide if $x \in \text{cl}(A \cup \bigcup_{j \in Q} S_j)$,
- (3) for all Q , and $i \notin Q$, $S_i \cap \text{cl}(A \cup \bigcup_{j \in B} S_j) = \emptyset$,
- (4) for all Q with $\text{card}(\omega - Q) = \infty$, $\text{cl}(A \cup \bigcup_{i \in Q} S_i) \neq^* M$,
- (5) for any finite Q , $i \notin Q$, and any finite set $\{v_1, \dots, v_k\}$,
 if $\text{cl}(v_1, \dots, v_k, f(i)) \cap \text{cl}(A \cup \bigcup_{j \in Q} S_j) \neq \text{cl}(\emptyset)$,
 then $\text{cl}(v_1, \dots, v_k) \cap \text{cl}(A \cup (\bigcup_{j \in Q} S_j) \cup \{f(i)\}) \neq \text{cl}(\emptyset)$.

Theorem 2.30 (Remmel [82]) *Suppose there is a recursive function f , and $a \in L(\mathcal{M})$ such that f is special over A . Then M contains a creative substructure V .*

We do not give the proof, but refer the reader to [82, Theorem 6.2]. Inspection of this proof reveals that if we can arrange a recursive s -set B for M over A such that $\forall i (S_i \subseteq B)$ so that, in addition, we can ensure that V

is generated by a creative subset of B . This applies to most of the examples such as vector spaces, Boolean algebras, orderings, etc..

On the other hand if $A \subseteq B$ and $\text{cl}_D(A)$ is creative, then it is not necessary for A to be creative. For example, it is easy to modify Remmel's result (Theorem 20 of [70]) to show the following result. (We leave the details to the reader.)

Theorem 2.31 *Let $\mathcal{M}$ be an effective Steinitz algebra satisfying $(*)$ below. Then if B is a recursive basis of M , there is a simple subset A of B such that $\text{cl}(A)$ is creative.*

- $(*)$ *If I is independent and infinite, and $n \in \omega$, there exists $x \in \text{cl}(I)$ such that $\text{card}(\text{supp}_I(x)) \leq n$.*

Downey [26] obtained several interesting results on orbits of creative subspaces. Not only does the analogue of Myhill's [68] theorem fail, but there are infinitely many non-elementary equivalent creative subspaces. For example, in [26], Downey defined $V \in L(V_\infty)$ to be *creative of type n* ($n \in \omega$) if V is creative, and there is a decidable subspace Q of V_∞ such that $V \subseteq Q$, $\dim(V_\infty/Q) = n$ and, for all decidable subspaces $W \supseteq V$, $W \supseteq Q$. From [26], we have

Theorem 2.32 (Downey)

- (i) *For each $n \in \omega$, there exists a subspace which is creative of type n .*
- (ii) *There are creative subspaces which are not creative of type n for any n*
- (iii) *Furthermore, for any n , there are infinitely many nonautomorphic subspaces all of which are creative of type n .*

We refer the reader to [26] for further details.

2.5 Speedable and Levelable Structures

In this section, we consider the speedable and levelable sets introduced by Blum and Marques [10] in our setting. In this section, we shall write $\overline{A}$ for $\omega - A$, and write $A \subset_\infty B$ if $A \subseteq B$ and $|B - A| = \infty$.

Let $\{\Phi_i : i \in \omega\}$ be step counting functions which constitute a complexity measure in the sense of Blum [9]. Namely, assume that:

- (1) $\varphi_i(x)$ converges if and only if $\Phi_i(x)$ converges, and
- (2) the function

$$M(i, x, y) = \begin{cases} 1 & \text{if } \Phi(x) = y \\ 0 & \text{otherwise} \end{cases}$$

is total recursive.

For example, $\Phi_i(x)$ could be the number of steps used by the i -th Turing machine P_i on input x , or the number of cells scanned (provided that $\Phi_i(x)$ is undefined if P_i does not halt on input x). We will write $\{e\}_s^A(x) \cong \{e\}_t^B(x)$ to mean that both computations are identical, namely $\{e\}_s^A(x) = \{e\}_t^B(x)$, and A_s and B_t are identical below the use $u(A_s; e, x, s)$ of this computation. We will use the symbols " $\exists^\infty x$ " and " $\forall^\infty x$ " to mean "there exist infinitely many x " and "for all but finitely many x " respectively.

Definition 2.33 An r.e. set A is *speedable* if for all i such that $W_i = A$ and for all recursive functions h , there exists a j such that $W_j = A$, and

$$(\exists^\infty x) [\Phi_i(x) > h(x, \Phi_j(x))]$$

Furthermore, we say that A is *effectively speedable* if j can be recursively obtained from i and an index for h .

Definition 2.34 An r.e. set A is *levelable* if there is a recursive function r such that, for all i with $W_i = A$ and for all recursive functions h , there exists a j such that $W_j = A$, and

$$(\exists^\infty x) [\Phi_i(x) > h(x) \text{ and } \Phi_j(x) < r(x)]$$

We say that A is *effectively levelable* if j can be recursively obtained from i and an index for h .

It is easy to see from the definitions that every levelable set is speedable and no recursive set can be speedable.

Intuitively, if A is speedable, there is no fastest program which computes A . That is, for every program P_i which computes A in the sense that $A = \{x : \varphi_i(x) \downarrow\}$, and for every recursive function h , there is another program P_j for A which is an h -speed-up of P_i for infinitely many x . Here we

say that P_j is an h -speed-up of P_i on argument x if $\Phi_i(x) > h(x, \Phi_j(x))$. The drawback of the notion of speedable from the computational complexity point of view is that even if, for every program P_i for A there is another program P_j for A which runs much faster than P_i on infinitely many inputs x , the running time $\Phi_j(x)$ may still be extremely large. The notion of P_j leveling P_i on an input x , is that not only does P_j run much faster than P_i on x , but in addition $\Phi_j(x) < r(x)$, where r is some predetermined recursive function. These properties have been widely studied by Blum, Marques, Gill, Morris, Soare, Bennison, Filotti, and others. Since the notions of speedable and levelable sets are not as well known as those of maximal, simple, creative sets considered earlier in this section, we will take some time to summarize a few of the main results on these types of sets. We start with some alternative characterizations of effectively speedable and effectively levelable sets.

Definition 2.35 (Blum and Marques [10]) An r.e. set is subcreative if and only if there exists a recursive function σ , with the property that for every index j , there is a natural number x such that $W_{\sigma(j)} = A \cup \{x\}$, where $x \in W_j \cap A$ or $x \in \overline{W_j} \cup \overline{A}$.

This was the original definition of subcreative as introduced in [10]. Blum and Marques then gave a characterization of subcreative sets which was strengthened by Gill and Morris [38] to the following.

Theorem 2.36 (Gill and Morris [38]) An r.e. set A is subcreative if and only if there exists a recursive function δ such that for all j , if $W_j \cap A = \emptyset$, then $A \subseteq W_{\delta(j)} \subseteq \overline{W_j}$.

Definition 2.37 (Filotti [35]) An r.e. set A is undercreative if and only if there exist recursive functions δ and δ' such that

- (1) $\forall j [W_{\delta(j)} = \overline{W_{\delta'(j)}}]$
- (2) $(W_j \cap A \text{ finite}) \rightarrow (W_{\delta(j)} \subseteq \overline{W_j - A} \text{ and } W_{\delta(j)} \cap \overline{W_j} \cup \overline{A} \neq \emptyset)$

Theorem 2.38 (Blum and Marques [10]) An r.e. set A is effectively speedable if and only if A is subcreative.

Theorem 2.39 (Filotti [35]) An r.e. set A is effectively levelable if and only if A is undercreative.

It is immediately clear from the definitions that every undercreative set is subcreative and thus every effectively levelable set is also effectively speedable. It is clear from the characterization of subcreative by Gill and Morris that every creative set is subcreative. Next we shall consider alternative characterizations of speedable and levelable sets.

Definition 2.40 Given a set A ,

- (i) define the *jump* of A as $A' = \{e : e \in W_e^A\}$, and the n -th *jump* of A to be $A^{(n)} = (A^{(n-1)})'$,
- (ii) define the *weak jump* of A as $A^{(1)} = H_{\bar{A}} = \{e : W_e \cap \bar{A} \neq \emptyset\}$, and the n -th *weak jump* of A to be $A^{(n)} = H_{\overline{A^{(n-1)}}} = \{i : W_i \cap \overline{A^{(n-1)}} \neq \emptyset\}$.
- (iii) A is *low* if $A' \leq_T \emptyset'$, and is *semilow* if $A^{(1)} = H_{\bar{A}} \leq_T \emptyset'$.
- (iv) In general, we say that A is *weak low_n* if $A^{(n)} \equiv_1 \emptyset^{(n)}$, and *strong high_n* if $A^{(n)} \equiv_1 \emptyset^{(n+1)}$.

Blum and Marques [10] gave a recursion theoretic characterization of non-speedable r.e. sets. However, for our purposes, we shall use an exceptionally neat characterization of nonspeedable sets due to Soare.

Theorem 2.41 (Soare [94]) *Let A be an r.e. set, then A is nonspeedable if and only if $\{e : W_e \cap \bar{A} \neq \emptyset\} \leq_T \emptyset'$ (i.e., A is semilow).*

Definition 2.42 A sequence $\mathcal{R} = \{R_n\}_{n \in \omega}$ of recursive sets is a

- (1) *recursively enumerable array* if there is a recursive function f such that $R_n = W_{f(n)}$.
- (2) *uniformly recursive array* if there is a recursive function g such that $\varphi_{g(n)}$ is the characteristic function of R_n for all n . If $g = \varphi_e$, then we say e is an *index* for $\mathcal{R}$.
- (3) *cofinal in an r.e. set A* if for all recursive sets B ,

$$B \supseteq \bar{A} \Rightarrow \exists n (R_n \subseteq A \ \& \ |R_n \cap B| = \infty).$$

Theorem 2.43 (Blum and Marques [10]) *An r.e. set A is levelable if and only if there is a uniformly recursive array $\{R_n\}_{n \in \omega}$ which is cofinal in A .*

Theorem 2.44 (Soare [94]) *Let A be an r.e. set, then A is nonlevelable if and only if $\{n : R_n \cap \bar{A} \neq \emptyset\} \leq_T \emptyset'$ for every uniformly recursive array $\mathcal{R} = \{R_n\}_{n \in \omega}$.*

Blum and Marques [10] also introduced the notion of an effectively non-levelable set.

Definition 2.45 An r.e. set is *effectively nonlevelable* if and only if there is a recursive function f such that for all i and k , if $W_i = A$ and φ_k is recursive, then $\varphi_{f(i,k)}$ is recursive, and

$$(\forall j) [(W_j = W_i) \Rightarrow (\forall^\infty x) ((\Phi_j(x) < \Phi_k(x)) \Rightarrow (\Phi_i(x) < \varphi_{f(i,k)}(x)))].$$

(Think of φ_k as corresponding to r and $\varphi_{f(i,k)}$ as corresponding to h , so that $\varphi_{f(i,k)}$ is an effective counterexample to A being levelable via φ_k .)

Theorem 2.46 *Let A be an r.e. set. Then the following are equivalent.*

- (1) A is effectively nonlevelable.
- (2) (Blum and Marques [10]) *There is a recursive function h such that, if e is an index of a uniformly recursive array $\{R_n\}_{n \in \omega}$, then $\varphi_{h(e)}$ is the characteristic function of a recursive set $B \supset \bar{A}$ satisfying*

$$(\forall n) ((R_n \subseteq A) \Rightarrow (R_n \cap B \text{ is finite}))$$

- (3) (Soare [94]) *There is a recursive function k such that, if e is an index of a uniformly recursive array $\{R_n\}_{n \in \omega}$, then $\{R_n \cap \bar{A} \neq \emptyset\} \leq_T \emptyset'$ by the Turing reduction of index $k(e)$.*

Bennison extended the work of Soare by giving information content characterizations for the effectively speedable and effectively levelable sets. Let $FIN = \{e : W_e \text{ is finite}\}$.

Theorem 2.47 (Bennison, [8]) *An r.e. set A is effectively speedable if and only if $A^{(1)} \equiv_1 \emptyset^{(2)}$ (i.e., A is strong high₁).*

Theorem 2.48 (Bennison, [8]) *An r.e. set A is effectively levelable if and only if there exists a uniformly recursive array $\{R_i\}_{i \in \omega}$ such that*

$$\{i : R_i \cap \bar{A} \neq \emptyset\} \equiv_1 FIN.$$

This given, the question that we ask in this section is, what is the relation between the speedability (levelability, etc.) of a generating set A and its closure $\text{cl}(A)$ in an effective closure system. This type of question was first studied by B  uerle and Remmel [6] in V_∞ .

We start with two basic results which are easily derived from Soare's characterization of nonspeedable and levelable sets, Theorems 2.41 and 2.44.

Theorem 2.49 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let $V \supseteq D$ be an r.e. substructure of $\mathcal{M}$ such that $D \neq^* V$, and B be a special set for V over D . Then*

- (a) *If B is speedable, then $\text{cl}(B)$ is speedable, and*
- (b) *If B is levelable, then $\text{cl}(B)$ is levelable.*

Proof. We shall show that if $\text{cl}(B)$ is nonspeedable then B is nonspeedable, and if $\text{cl}(B)$ is nonlevelable then B is nonlevelable.

The key observation is that $\text{cl}(B) - B$ is r.e.. That is, for each $x \in \text{cl}(B)$, we can effectively compute $\text{supp}_B(x)$. Thus $x \in \text{cl}(B) - B$ if and only if either $\text{card}(\text{supp}_B(x)) \geq 2$ or $\text{supp}_B(x) = \{b\}$, where $b \in B$ and $x \neq b$. Thus, for any r.e. set W_e , $W_e \cap \overline{B} \neq \emptyset$ if and only if either $W_e \cap \overline{\text{cl}(B)} \neq \emptyset$ or $W_e \cap (\text{cl}(B) - B) \neq \emptyset$. Thus it follows that, if $\{e : W_e \cap \overline{\text{cl}(B)} \neq \emptyset\} \leq_T \emptyset'$, then $\{e : W_e \cap \overline{B} \neq \emptyset\} \leq_T \emptyset'$.

Similarly, for any uniform recursive array $\mathcal{R} = \{R_n\}_{n \in \omega}$, if

$$\{e : R_e \cap \overline{\text{cl}(B)} \neq \emptyset\} \leq_T \emptyset',$$

then

$$\{e : R_e \cap \overline{B} \neq \emptyset\} \leq_T \emptyset'.$$

Hence B is nonlevelable if $\text{cl}(B)$ is nonlevelable. □

Moreover, if B is a subset of special set S for M over D , where $M = \omega$, then we can say a bit more.

Theorem 2.50 *Let $\mathcal{M} = (\omega, \text{cl})$ be an effective closure system, and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let S be a special set for M over D , and B be an r.e. subset of S . Then B is speedable if and only if $\text{cl}(B)$ is speedable.*

Proof.

By Theorem 2.49, if B is speedable, then $\text{cl}(B)$ is speedable.

Now suppose that $\text{cl}(B)$ is speedable. Then it is easy to see that $W_e \cap \text{cl}(B) \neq \emptyset$ if and only if $\text{supp}_S(W_e) \cap B \neq \emptyset$. Since there is a recursive function g such that $\text{supp}_S(W_e) = W_{g(e)}$, it follows that

$$\emptyset' <_T \{e : W_e \cap \overline{\text{cl}(B)} \neq \emptyset\} \leq_T \{e : W_e \cap \overline{B} \neq \emptyset\},$$

so that B is speedable. □

We note that it is not the case that under the hypothesis of Theorem 2.50 that if $\text{cl}(B)$ is levelable, then B is levelable. That is, Bäuerle and Rimmel [6] proved that in V_∞ , any subspace generated by a speedable subset of recursive basis for V_∞ is levelable. Moreover Blum and Marques [10] proved that there is an effectively nonlevelable effectively speedable set. Thus if we take an effectively nonlevelable speedable subset of a recursive basis of V_∞ , it will generate a subspace which is levelable.

Theorem 2.51 *Let $\mathcal{M} = (\omega, \text{cl})$ be an effective closure system and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let $V \supseteq D$ be an r.e. substructure of $\mathcal{M}$ such that $D \neq^* V$, and B be a special set for V over D . Then*

- (a) *if B is effectively speedable, then $\text{cl}(B)$ is effectively speedable, and*
- (b) *if B is effectively levelable, then $\text{cl}(B)$ is effectively levelable.*

Proof.

(a) To show that $V = \text{cl}(B)$ is effectively speedable, we must show that there exists a recursive function α such that if $W_j \cap V = \emptyset$, then

$$V \subset W_{\alpha(j)} \subseteq \overline{W_j}.$$

Now because B is effectively speedable, there is a recursive function δ such that if $W_j \cap B = \emptyset$, then $A \subset W_{\delta(j)} \subseteq \overline{W_j}$. Now note that $\text{cl}(B) - B$ is r.e., so there is a recursive function f such that $W_{f(j)} = W_j \cup (\text{cl}(B) - B)$. Similarly, there is a recursive function g such that $W_{g(j)} = W_j \cup \text{cl}(B)$. We claim that our desired function α is given by $\alpha = g \circ \delta \circ f$. That is, suppose

that $W_j \cap \text{cl}(B) = \emptyset$, then $W_{f(j)} = W_j \cup (\text{cl}(B) - B)$; and since $B \subseteq \text{cl}(B)$, then also $W_{f(j)} \cap B = \emptyset$. Thus,

$$B \subset W_{\delta(f(j))} \subseteq \overline{W_j \cup (\text{cl}(B) - B)}.$$

In particular,

$$(W_{\delta(f(j))} - B) \cap \overline{\text{cl}(B)} \neq \emptyset.$$

Moreover, since $W_j \cap \text{cl}(B) = \emptyset$,

$$W_{g(\delta(f(j)))} = W_{\delta(f(j))} \cup \text{cl}(B) \subseteq \overline{W_j}.$$

Thus $\text{cl}(B) \subset W_{\alpha(j)} \subseteq \overline{W_j}$ as desired.

(b) Since B is effectively levelable, it follows from Theorem 3.39 that there exist recursive functions δ and δ' such that for all j

$$(1) \quad \forall j \ (W_{\delta(j)} = \overline{W_{\delta'(j)}}), \text{ and}$$

$$(2) \quad (W_j \cap B \text{ is finite}) \rightarrow (W_{\delta(j)} \subseteq \overline{W_j - B} \text{ and } W_{\delta(j)} \cap \overline{W_j \cup B} \neq \emptyset).$$

Now let $\gamma(j) = \delta(f(j))$ and $\gamma'(j) = \delta'(f(j))$, where $W_{f(j)} = W_j \cup (\text{cl}(B) - B)$. We claim

$$(i) \quad \forall j \ (W_{\gamma(j)} = \overline{W_{\gamma'(j)}}), \text{ and}$$

$$(ii) \quad (W_j \cap \text{cl}(B) \text{ is finite}) \\ \rightarrow (W_{\gamma(j)} \subseteq \overline{W_j - \text{cl}(B)} \text{ and } W_{\gamma(j)} \cap \overline{W_j \cup \text{cl}(B)} \neq \emptyset),$$

so that $\text{cl}(B)$ is also effectively levelable. First, condition (i) is immediate. For condition (ii), note that if $W_j \cap \text{cl}(B)$ is finite, then

$$W_{f(j)} \cap B = (W_j \cup (\text{cl}(B) - B)) \cap B$$

is also finite. Thus,

$$W_{\gamma(j)} = W_{\delta(f(j))} \subseteq \overline{(W_j \cup (\text{cl}(B) - B)) - B} \subseteq \overline{W_j - \text{cl}(B)}.$$

Moreover,

$$\begin{aligned} W_{\gamma(j)} \cap \overline{W_j \cup \text{cl}(B)} &= W_{\gamma(j)} \cap \overline{W_j \cup (\text{cl}(B) - B) \cup B} \\ &= W_{\delta(f(j))} \cap \overline{W_{f(j)} \cup B} \neq \emptyset. \end{aligned}$$

□

Theorem 2.52 *Let $\mathcal{M} = (\omega, \text{cl})$ be an effective closure system and D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Let S be a special set for M over D , and B be an r.e. subset of S . Then B is effectively speedable if and only if $\text{cl}(B)$ is effectively speedable.*

Proof. By Theorem 2.51, we know that if B is effectively speedable, then $\text{cl}(B)$ is effectively speedable. Now suppose that $\text{cl}(B)$ is effectively speedable. To show B is effectively speedable, it is enough by Theorem 2.38 to show that B is subcreative with respect to S . That is, we must show that there is a recursive function δ such that if $W_j \subseteq S$ and $W_j \cap B = \emptyset$, then $B \subset W_{\delta(j)} \subseteq S - W_j$. Let α be the recursive function such that if $W_j \cap \text{cl}(B) = \emptyset$, then $\text{cl}(B) \subset W_{\alpha(j)} \subseteq \overline{W_j}$. Note, that for any r.e. set W_j , $N(W_j) = \{x \mid \text{supp}_S(x) \cap W_j \neq \emptyset\}$ is an r.e. set. In fact, there is a recursive function r such that $W_{r(j)} = N(W_j)$. Moreover, if $W_j \subseteq S$ and $W_j \cap B = \emptyset$, then $N(W_j) \cap \text{cl}(B) = \emptyset$. Similarly, for any r.e. set W_j , $\text{supp}_S(W_j)$ is an r.e. subset of S , and there is a recursive function t such that $W_{t(j)} = \text{supp}_S(W_j)$. We claim our desired function δ is given by $t \circ \alpha \circ r$. That is, suppose $W_j \subseteq S - B$. Then $W_{r(j)} = N(W_j)$ does not intersect $\text{cl}(B)$, so that $\text{cl}(B) \subset W_{\alpha(r(j))} \subseteq \overline{N(W_j)}$. But note $\overline{N(W_j)} = \{x \mid \text{supp}_S(x) \cap W_j = \emptyset\}$, so that $\text{supp}_S(W_{\alpha(r(j))}) \subseteq S - W_j$. Moreover, since $W_{\alpha(r(j))} - \text{cl}(B) \neq \emptyset$, there is a $y \in W_{\alpha(r(j))}$ such that $\text{supp}_S(y) \cap (S - B) \neq \emptyset$. Since $\text{cl}(B) \subseteq W_{\alpha(r(j))}$, $B \subseteq \text{supp}_S(W_{\alpha(r(j))}) = W_{\delta(j)}$. Thus $B \subset W_{\delta(j)} \subseteq S - W_j$ as desired. $\square$

We note that it is not the case that under the hypothesis of Theorem 2.52 that if $\text{cl}(B)$ is effectively levelable, then B is effectively levelable. That is, Bäuerle and Remmel [6] proved that in V_∞ , any subspace generated by an effectively speedable subset of a recursive basis for V_∞ is effectively levelable. Thus we can again use the result of Blum and Marques [10] that there is an effectively nonlevelable effectively speedable set. Thus if we take an effectively nonlevelable effectively speedable subset of a recursive basis of V_∞ , it will generate a subspace which is effectively levelable.

3 Sacks' splittings

In this section we shall investigate the analogue of Sacks' splitting theorem, which states that if A is a nonrecursive r.e. set, then there exist disjoint r.e. sets B and C such that $A = B \cup C$ and $B \upharpoonright_T C$.

We note that the standard proof of Sacks' splitting theorem uses the fact that if A and B are disjoint r.e. sets, then the Turing degrees of A and B join up to the Turing degree of C , i.e., $\deg(A) \vee \deg(B) = \deg(A \cup B)$. Thus in particular, $A \leq_T (A \cup B)$ and $B \leq_T (A \cup B)$. It is not true in V_∞ that if A and B are r.e. subspaces such that $A \cap B = \text{cl}(\emptyset)$, then $\deg(A) \vee \deg(B) = \deg(\text{cl}(A \cup B))$. For example, Ash and Downey [1] showed that any r.e. subspace of V_∞ is a direct sum of two decidable subspaces. It is the case however, that if A and B are r.e. subspaces such that $A \cap B = \text{cl}(\emptyset)$, then $A \leq_T (\text{cl}(A \cup B))$ and $B \leq_T (\text{cl}(A \cup B))$. In Boolean algebras, the situation is even worse. For example, Yang, in his thesis [97], showed that for any two r.e. degrees α and β , there are r.e. subalgebras of $\tilde{N}$, A of degree α and B of degree β , such that $A \cap B = \{0_{\tilde{N}}, 1_{\tilde{N}}\}$ yet $\text{cl}(A \cup B) = \tilde{N}$. In fact, similar examples exist for all Boolean algebras of the form $\tilde{N} \times D$, $\tilde{Q} \times D$, and $\tilde{C}$, where D is any recursive Boolean algebra.

Also, it is not true that if $V \in L(V_\infty)$, and if B is an r.e. basis of V , and if B_1 and B_2 are disjoint r.e. sets which form a Sacks' splitting of B (i.e., $B_1 \upharpoonright_T B_2$ and $B_1 \cup B_2 = B$), then $(B_1)^* \upharpoonright_T (B_2)^*$. Indeed, as is shown in [32], we may have $(B_2)^* \equiv_T (B_1)^*$ although $B_1 \upharpoonright_T B_2$. Thus Sacks splittings of special sets do not induce Sacks splitting of substructures.

We can give a direct construction which will produce Sacks' splittings of substructures which are generated by non-recursive r.e. special sets with LEP. Unfortunately, this does not imply the existence of Sacks' splittings of all r.e. substructures of an effective closure system. For example, Yang [97] showed for that any r.e. subalgebra A of $\tilde{Q}$ that has a nonrecursive r.e. generating sequence, i.e., a sequence $a_0, a_1, \dots$ such that $A = \text{cl}(\{a_0, a_1, \dots\})$ and for all $n > 0$, a_n is strictly contained in some atom of $\text{cl}(a_0, \dots, a_{n-1})$, then there exist r.e. subalgebras A_1 and A_2 of $\tilde{Q}$ such that $A_1 \cap A_2 = \{0_{\tilde{Q}}, 1_{\tilde{Q}}\}$, $\text{cl}(A_1 \cup A_2) = A$, and $A_1 \upharpoonright_T A_2$. However, it is not known whether every r.e. subalgebra of $\tilde{Q}$ has a non-recursive r.e. generating sequence. We note that proving a Sacks' splitting result for r.e. subalgebras of $\tilde{Q}$ proves a Sacks' splitting result for all r.e. subalgebras of any recursive Boolean algebra $\mathcal{B}$, since Remmel proved that every recursive Boolean algebra $\mathcal{B}$ is recursively isomorphic to a recursive subalgebra of $\tilde{Q}$. Yang [97] did prove that every nonrecursive r.e. subalgebra of $\tilde{N}^k$ has a nonrecursive r.e. generating sequence. We note that an r.e. generating sequence is not necessarily a special sequence. Finally, we shall see in Section 5 that the analogue of the Sacks' splitting theorem simply fails in $(V_\infty, K\ell)$.

We will derive our Sacks splitting theorem for effective closure systems from the following result.

Theorem 3.1 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and let D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Suppose that $V \supseteq D$ is an r.e. substructure of $\mathcal{M}$, and G is a special set for V over D . Let C be a nonrecursive r.e. subset of ω . Then there exist r.e. sets G_0 and G_1 which split G such that if $B_i = \text{cl}_D(G_i)$ for $i = 0, 1$, then*

- (i) $V = B_0 + B_1$,
- (ii) $B_0 \cap B_1 = D$, and
- (iii) $C \not\leq_T B_i$, for $i = 0, 1$.

Proof.

Given a set $B \subseteq \omega$, we let $B \downarrow r = \{x \leq r : x \in B\}$.

Let $g_0, g_1, \dots$ be an r.e. enumeration of G without repetitions. Let $G_s = \{g_0, \dots, g_s\}$ for all s . Let $\{C_s\}_{s \in \mathbb{N}}$ be an effective enumeration of C . We'll enumerate $\{G_{i,s}\}_{\{s \in \omega\}}$ and $\{B_{i,s}\}_{\{s \in \omega\}}$ for $i = 0, 1$, where $B_{i,s} = \text{cl}_D(G_{i,s})$ for all i and s , so as to satisfy the positive requirement

$P : g_s$ is put into exactly one of $G_{0,s}$ or $G_{1,s}$

and the negative requirement, for $i = 0, 1$ and all e ,

$$N_{\langle e, i \rangle} : C \neq \{e\}^{B_i}.$$

Note that meeting the positive requirement immediately ensures both conditions (i) and (ii) are satisfied.

We'll construct B_0 and B_1 in stages. First we order all pairs $\langle e, i \rangle$, where $e \in \omega$ and $i \in \{0, 1\}$, lexicographically, i.e., $\langle e, i \rangle < \langle f, j \rangle$ if either $e < f$ or both $e = f$ and $i < j$.

Construction

STAGE 0:

Define $G_{0,0} = \{g_0\}$ and $G_{1,0} = \emptyset$.

STAGE $s + 1$:

Having defined $G_{i,s}$ and $B_{i,s}$, define recursive functions $l^i(e, s)$ and $r^i(e, s)$

(length function) $l^i(e, s) = \max\{x : (\forall y < x) \{e\}_s^{B_{i,s}}(y) \downarrow = C_s(y)\}$,

(restraint function) $r^i(e, s) = \max\{u(B_{i,s}; e, x, s) : x \leq l(e, s)\}$.

Let $p_i = \mu x(x \in \text{cl}_D(B_{i,s} \cup \{g_{s+1}\}) - B_{i,s})$. Choose $\langle e', i' \rangle$ to be the least $\langle e, i \rangle$ such that $p_i \leq r^i(e, s)$, then enumerate g_{s+1} into $G_{1-i',s+1}$. That is, let

$$G_{i',s+1} = G_{i',s}, \quad B_{i',s+1} = B_{i',s},$$

$$G_{1-i',s+1} = G_{1-i',s} \cup \{g_{s+1}\}, \quad B_{1-i',s+1} = \text{cl}_D(G_{1-i',s+1}).$$

(Note $N_{\langle e', i' \rangle}$ is the highest priority requirement which might be injured by enumerating g_{s+1} in $B_{i'}$, so we enumerate g_{s+1} into $B_{1-i'}$.) If no such $\langle e', i' \rangle$ exists, enumerate g_{s+1} into G_0 . That is, let

$$G_{0,s+1} = G_{0,s} \cup \{g_{s+1}\}, \quad B_{0,s+1} = \text{cl}_D(G_{0,s+1}),$$

$$G_{1,s+1} = G_{1,s}, \quad B_{1,s+1} = B_{1,s}.$$

To see that the construction succeeds, we define the injury set

$$(\text{injury set}) \quad I_e^i = \{x : (\exists s)[x \in B_{i,s+1} - B_{i,s} \ \& \ x \leq r(e, s)]\}.$$

We can prove by induction on $\langle e, i \rangle$ that for $i = 0, 1$ and all e ,

- (1) I_e^i is finite,
- (2) $C \neq \{e\}^{B_i}$, and
- (3) $r^i(e) = \lim_s r^i(e, s)$ exists and is finite.

Namely, fix $\langle e, i \rangle$, and assume (1), (2) and (3) hold for all $\langle k, j \rangle < \langle e, i \rangle$. By (3), choose t such that $r^j(k, s) = r^j(k)$ for all $\langle k, j \rangle < \langle e, i \rangle$ and all $s \geq t$.

Choose r greater than all such $r^j(k)$, and then choose $v > t$ such that $V \downarrow r = \text{cl}_D(G_v) \downarrow r$. Now $N_{\langle e, i \rangle}$ is never injured after stage v . This is because otherwise, suppose that there is a stage $w \geq v$ at which $N_{\langle e, i \rangle}$ is injured. But then by the construction, there exists $\langle k, 1-i \rangle < \langle e, i \rangle$ such that if

$$p_i = \mu x(x \in \text{cl}_D(B_{i,w} \cup \{g_{w+1}\}) - B_{i,w}),$$

then $p_i \leq r^{1-i}(k, s) < r$. But then $p_i \in \text{cl}_D(G_{w+1}) \downarrow r - \text{cl}_D(G_w) \downarrow r$, contrary to the hypothesis that $V \downarrow r = \text{cl}_D(G_v) \downarrow r$. That is, because G is a special set for V ,

$$p_i \in \text{cl}_D(B_{i,w} \cup \{g_{w+1}\}) - B_{i,w} = \text{cl}_D(G_{i,w} \cup \{g_{w+1}\}) - \text{cl}_D(G_{i,w})$$

implies $g_{w+1} \in \text{supp}_G(p_i)$, and hence $p_i \notin \text{cl}_D(G_w)$. Thus (1) holds for $\langle e, i \rangle$.

We note that (2) and (3) hold for $\langle e, i \rangle$ by essentially the same arguments given in Theorem 3.2 (Sacks' splitting Theorem) in Soare [92]. That is, to see that $C \neq \{e\}^{B_i}$, assume for a contradiction that $C = \{e\}^{B_i}$ for some $i = 0$ or 1 . Then $\lim_s l^i(e, s) = \infty$. By (1), choose s' such that $N_{\langle e, i \rangle}$ is never injured after stage s' . We shall recursively compute C contrary to hypothesis. To compute $C(p)$ for some $p \in \omega$, find the least $s > s'$ such that $l^i(e, s) > p$. It follows by induction on $t \geq s$ that

$$(\forall t \geq s) (l^i(e, t) > p \ \& \ r^i(e, t) \geq \max\{u(B_{i,s}; e, x, s) : x \leq p\}), \quad (*)$$

and hence that $\{e\}_s^{B_{i,s}}(p) = \{e\}^{B_{i,s}}(p) = \{e\}_i^B(p) = C(p)$. Hence C would be recursive, contrary to our hypothesis. To prove $(*)$, assume that it holds for t . Then because of the definition of $r^i(e, t)$ and the fact that $t > s'$, $B_{i,t+1} \downarrow z = B_{i,t} \downarrow z$ for all numbers z used in the computation $\{e\}_t^{B_{i,t}}(x) = y$, for any $x \leq p$. Thus

$$\{e\}_{t+1}^{B_{i,t+1}}(x) \downarrow = \{e\}_t^{B_{i,t+1}}(x) = \{e\}_t^{B_{i,t}}(x) = y,$$

so $l^i(e, t+1) > p$, unless $C_{t+1}(x) \neq C_t(x)$ for some $x < l^i(e, t)$. But if $C_t(x) \neq C_s(x)$, for some $t \geq s$ and $x \leq p$ where x is minimal, then our use of " $\leq l^i(e, t)$ " rather than " $< l^i(e, t)$ " in the definition of $r^i(e, t)$, ensures that the disagreement $C_t(x) \neq \{e\}_t^{B_{i,t}}(x)$ is preserved forever, contrary to our hypothesis that $C = \{e\}^{B_i}$. To show (3), i.e., that $(\forall e) [\lim_s r^i(e, s) \text{ exists and is finite}]$, by (1) and (2), choose $p = (\mu x)[C(x) \neq \{e\}^{B_i}(x)]$. Choose s' sufficiently large such that for all $s \geq s'$,

$$(1) (\forall x < p) [\{e\}_s^{B_{i,s}}(x) \downarrow = \{e\}^{B_i}(x)],$$

$$(2) (\forall x \leq p) [C_s(x) = C(x)], \text{ and}$$

$$(3) N_{\langle e, i \rangle} \text{ is not injured at stage } s.$$

CASE 1: $(\forall s \geq s') [\{e\}_s^{B_{i,s}}(p) \uparrow]$. Then $r^i(e, s) = r^i(e, s')$ for all $s \geq s'$.

CASE 2: $\{e\}_t^{B_{i,t}}(p) \downarrow$ for some $t \geq s'$. Then $\{e\}_t^{B_{i,t}}(p)$ for all $s \geq t$ because $l^i(e, s) \geq p$, and so, by the definition of $r^i(e, s)$, and the fact that $N_{\langle e, i \rangle}$ is not injured after stage t , the computation $\{e\}^{B_{i,t}}(p)$ is preserved forever. Thus $\{e\}^{B_i}(p) = \{e\}_s^{B_{i,s}}(p)$. But $C(p) \neq \{e\}^{B_i}(p)$. Hence

$$(\forall s \geq t) (C_s(p) \neq \{e\}_s^{B_{i,s}}(p) \ \& \ l^i(e, s) = p \ \& \ r^i(e, s) = r^i(e, t)).$$

Therefore, $r^i(e, t) = \lim_s r^i(e, s)$. □

Corollary 3.2 *Let $\mathcal{M} = (M, \text{cl})$ be an effective closure system, and let D be a decidable substructure of $\mathcal{M}$ such that $D \neq^* M$. Suppose that $V \supseteq D$ is an r.e. substructure of $\mathcal{M}$, and G is a special set for V over D which is not recursive. Then there exist r.e. sets G_0 and G_1 which split G such that if $B_i = \text{cl}_D(G_i)$ for $i = 0, 1$, then*

$$(i) \quad V = B_0 + B_1,$$

$$(ii) \quad B_0 \cap B_1 = D, \text{ and}$$

$$(iii) \quad C \not\leq_T B_i \text{ for } i = 0, 1.$$

Proof. Since G is a nonrecursive r.e. special set for V , we can use Theorem 4.1 where $C = G$. That is, let G_0, G_1 and B_0, B_1 be constructed as in Theorem 4.1. Then $G \equiv_T G_0 \oplus G_1$, where $\oplus$ means set direct sum, i.e., $G_0 \oplus G_1 = \{2i : i \in G_0\} \cup \{2i + 1 : i \in G_1\}$. Note $G_i \leq_T B_i$, since G_i is clearly a special set for B_i . Hence $C = G \leq_T B_0 \oplus B_1$, where $\oplus$ is also the set direct sum. Therefore if $\deg(B_0)$ is comparable to $\deg(B_1)$, for example $B_0 \leq_T B_1$, then $C \leq_T B_1$, which is a contradiction. $\square$

That the complete analogue of Sacks' splitting holds in $L(V_\infty)$ is a result due to Downey, Remmel and Welch [32]. We remark that they established the theorem via a general result relating the degrees of bases of $V \in L(V_\infty)$ to the degrees of splittings. This result relies heavily on certain structural properties of $L(V_\infty)$.

The degrees of splittings and bases and how they relate to the models they generate all turn out to have a complex relationship even for V_∞ . We refer the reader to [32, 33, 97] for further details, and give only the following examples from [32].

Theorem 3.3 (Downey, Remmel and Welch) *Let $V \in L(V_\infty)$. Then δ is the degree of an r.e. basis of V if and only if there exist $W_1, W_2 \in L(V_\infty)$ such that $d(W_1) = d(D(W_1)) = \delta$ and $W_1 \oplus W_2 = V$.*

Theorem 3.4 (Downey, Remmel and Welch) *There exist $Q, V, W \in L(V_\infty)$ such that $Q \oplus V = W$ and $\emptyset <_T Q <_T W$, but such that for all $V', Q' \in L(V_\infty)$, if*

$$V'' \oplus Q' = W \text{ and } Q \equiv_T Q',$$

then

$$d(Q') \vee d(V') \neq d(W).$$

In point of fact, the degrees of bases and splitting are related more to weak truth table degrees than to Turing degrees. The following theorem is definitive.

Theorem 3.5 (Downey and Remmel [30]) *Suppose that A is recursively enumerable. Then V has a basis of weak truth table degree $\deg(A)$ iff $A \leq_{\text{wtt}} V$.*

4 Another example: $K(V_\infty)$

The previous section concerned itself with the situation of a result which held fairly generally, yet could not be lifted from $L(\omega)$ by a coding technique. In this section we shall analyze another lattice to which most of the results of Section 2 do not pertain, since the lattice doesn't obey the underlying assumptions involved in their proofs. The reader might feel that since our assumptions are so weak, all interesting lattices arising naturally would satisfy these assumptions. We shall see that this is not the case for the lattice $K(V_\infty)$. Recall from Section 1, that $K(V_\infty)$ consists of an infinite dimensional recursive vector space V_∞ over a recursively ordered recursive field F which is isomorphic to the rationals $\mathbb{Q}$. In addition, following Kalantari [46], we assume that V_∞ possesses

- (a) a *dependence algorithm*, i.e., we assume one can decide whether or not $x \in \{y_1, \dots, y_n\}^*$, where $\{y_1, \dots, y_n\}^*$ is the subspace generated by $\{y_1, \dots, y_n\}$, and
- (b) a *convexity algorithm*, i.e., we assume one can decide whether or not $\langle y_1, \dots, y_n \rangle \cap \langle z_1, \dots, z_m \rangle = \emptyset$, where $\langle y_1, \dots, y_n \rangle$ is the convex hull generated by $y_1, \dots, y_n$.

The original paper which introduced the study of the lattice of r.e. convex sets, $K(V_\infty)$, was [46]. $K(V_\infty)$ is obviously a very natural setting for studying the effective content of various geometric theorems such as the Hahn Banach theorem, Mazur's theorem, Stone's separation theorem, and others. It has since been further analyzed in [22, 24, 28, 48].

Actually, although Kalantari asked that V_∞ satisfy both (a) and (b) above, it suffices that it satisfy only one. That is, the following holds.

Lemma 4.1 (Nevins [73]) *Let V be an r.e. presented vector space over a recursively ordered recursive field. Then V has a dependence algorithm if and only if V has a convexity algorithm.*

In this section we will study several particular features of $K(V_\infty)$. Already we know that $K(V_\infty)$ is very complex. That is, the following was proved in [24]. Here $\text{Th}(A)$ denotes the first order theory of A .

Theorem 4.2 (Downey [24]) *$\text{Th}(K(V_\infty))$ is undecidable. In fact, we can effectively interpret $\text{Th}(L(V_\infty))$ and $\text{Th}(L(\omega))$ in $\text{Th}(K(V_\infty))$.*

We do not know if the related structure $K(V_\infty)/=^*$, that is, $K(V_\infty)$ factored out by the congruence $=^*$, is undecidable, although we conjecture that it is undecidable.

Our inability to apply the results of Section 2 is due the following.

Theorem 4.3 *There is no special set for V_∞ over $\emptyset$ in $K(V_\infty)$.*

Proof. Suppose that $S = \{s_0, s_1, \dots\}$ is a special set for V_∞ over $\emptyset$. Let k be the least j such that $\vec{0} \in \langle s_0, \dots, s_j \rangle$, and consider $\langle s_0, \dots, s_{k+1} \rangle$. There is some maximum $a \in F$ such that $as_{k+1} \in \langle s_0, \dots, s_{k+1} \rangle$. Clearly $a \geq 1$. Let $T = \text{supp}_S(2as_{k+1})$. There are two cases.

CASE 1: $s_{k+1} \notin T$. Then

$$s_{k+1} = \frac{1}{2a}(2as_{k+1}) + (1 - \frac{1}{2a})\vec{0},$$

which shows that $s_{k+1} \in \langle \{s_i : i \in \{0, \dots, k\} \cup T\} \rangle$. However $s_{k+1} \in \langle s_{k+1} \rangle$. Thus if S were special, we could conclude that

$$\text{supp}_S(s_{k+1}) \subseteq \{s_{k+1}\} \cap (\{0, \dots, k\} \cup T) = \emptyset,$$

which is impossible.

CASE 2: $s_{k+1} \in T$. In this case,

$$2as_{k+1} = \gamma s_{k+1} + (1 - \gamma)u,$$

where $0 \leq \gamma \leq 1$ and $\text{supp}_S(u) \subseteq T - \{s_{k+1}\}$. Note that $\gamma \neq 0$ since otherwise we are in case 1, and $\gamma \neq 1$ since $2a \geq 2$. But then

$$u = \frac{2a - \gamma}{1 - \gamma} s_{k+1}$$

has its support in $T - \{s_{k+1}\}$. Moreover it is easy to see that for $0 < \gamma < 1$,

$$\frac{2a - \gamma}{1 - \gamma} = 1 + \frac{2a - 1}{1 - \gamma} \geq 2a.$$

But then we can use the same argument as in case 1 to derive a contradiction. Thus there is no special set for V_∞ over $\emptyset$ in $K(V_\infty)$. $\square$

One of the key facts about $K(V_\infty)$ is that, if $V \in K(V_\infty)$ and $V =^* \emptyset$, then V is complemented in $K(V_\infty)$. Complementation is often the first point where an algebraic lattice differs from $L(\omega)$ and there is no exception here. Let $C(V_\infty)$ denote the set of complemented members of $K(V_\infty)$. Following [22] we say $V \in K(V_\infty)$ is *c-decidable* if, given any $x_1, \dots, x_n, y_1, \dots, y_m \in V_\infty$, uniformly we can decide whether or not

$$\langle V \cup \{x_1, \dots, x_n\} \rangle \cap \langle y_1, \dots, y_m \rangle = \emptyset.$$

The usual proof of Stone's separation theorem (e.g., [51]) shows that if $V \in K(V_\infty)$ is c-decidable, there exist $V', W \in K(V_\infty)$ such that $V' \cap W = \emptyset$, $V' \supset V$ and $V' \cup W = V_\infty$. Notice that any $V \in K(V_\infty)$ with $V =^* \emptyset$ is c-decidable and complemented, in the sense that there is $W \in K(V_\infty)$ with $\langle V \cup W \rangle = V_\infty$ and $V \cap W =^* \emptyset$. However, a decidable subspace V with $\dim(V_\infty/V) = \infty$ is c-decidable by Nevin's Theorem, but is not necessarily complemented in $K(V_\infty)$. This answers a question in [70], and shows that $K(V_\infty)$ differs substantially from $L(V_\infty)$. Indeed, we have the following.

Theorem 4.4 *Let $V \in L(V_\infty, K\ell)$. Then $V \in C(V_\infty)$ if and only if $V = \emptyset$ or $V = V_\infty$.*

Proof. Suppose $V \neq \emptyset$ and $V \neq V_\infty$, but $V \in L(V_\infty, K\ell) \cap C(V_\infty)$. We may suppose $\vec{0} \in V$, for otherwise we can apply the translation map $\vec{0} \rightarrow -v$, $x_i \rightarrow (x_i - v)$, where $v \in V$ and $\{x_i\}$ is a recursive basis of V_∞ . This obviously induces a recursive automorphism Φ of $K(V_\infty)$ by taking $\sum \alpha_i x_i \rightarrow \sum \alpha_i x_i - v$, and we may consider $\Phi(V)$ instead of V . Thus, without loss of generality, $\vec{0} \in V$. The advantage of having $\vec{0}$ in V is that if $V \in L(V_\infty, K\ell)$ and $\vec{0} \in V$, then V is also a subspace of V_∞ .

Let $C \in K(V_\infty)$, with $\langle C \cup V \rangle = V_\infty$ and $C \cap V = \emptyset$. Since $\vec{0} \in V$, it follows that for all $x \in V_\infty$ and $\alpha \in F$, $\alpha x \notin C$ or $-\alpha x \notin C$ (else $\vec{0} \in C$). Now choose $y \in C$ (since $C \neq \emptyset$). Thus $-y \in \langle V \cup C \rangle$, so that we can write

$$-y = \sum \alpha_i z_i + \sum \gamma_i c_i \tag{*}$$

where $0 \leq \alpha_i, \gamma_i \leq 1$, $\sum \gamma_i + \sum \alpha_i = 1$, for $z_i \in V$, $c_i \in C$. Notice $\sum \alpha_i \neq 0$ lest $-y \in C$. Also, $\sum \gamma_i c_i \neq \vec{0}$, since otherwise $-y \in V$, which in turn implies that $y \in V$ since $V \in L(V_\infty, K\ell)$. By $(*)$, it follows that

$$(\sum \alpha_i)^{-1}(-y - \sum \gamma_i c_i) \in V.$$

Now $(\sum \alpha_i)^{-1} > 1$ and $\vec{0} \in V$, hence $-y - \sum \gamma_i c_i \in V$. But then since $\vec{0} \in V$ and $V \in L(V_\infty, K\ell)$, $y + \sum \gamma_i c_i \in V$. Consequently we have

$$p = (\sum \gamma_i + 1)^{-1}y + \sum \gamma_j (\sum \gamma_i + 1)^{-1}c_i \in V,$$

$$0 \leq (\sum \gamma_i + 1)^{-1}, \gamma_j (\sum \gamma_i + 1)^{-1} \leq 1,$$

and

$$(\sum \gamma_i + 1)^{-1} + \sum \gamma_j (\sum \gamma_i + 1)^{-1} = (1 + \sum \gamma_i)(\sum \gamma_i + 1)^{-1} = 1.$$

But as $y \in C$ and $c_i \in C$, $p \in C$ because C is convex. Hence $p \in C \cap V$, a contradiction. $\square$

Using the result, we can also show

Theorem 4.5 $C(V_\infty)$ is not an upper semilattice.

Proof. Consider the rays (for $a \neq 0$),

$$H(a) = \langle \{na : n > 0 \text{ and } n \in \omega\} \rangle$$

$$H(-a) = \langle \{-na : n > 0 \text{ and } n \in \omega\} \rangle.$$

Obviously $H(a), H(-a) \in C(V_\infty)$. We claim $H(a) \vee H(-a)$ doesn't exist in $C(V_\infty)$. Let $D \in C(V_\infty)$ with $D = H(a) \vee H(-a)$. Then D contains the line $K\ell(\{0, a\})$. By Theorem 4.4 we know $D \neq K\ell(0, a)$. Thus, pick $x \in D - K\ell(\{0, a\})$. Then we can effectively apply Stone's separation theorem to the convex sets $A_1 = K\ell(\vec{0}, a)$ and $A_2 = \{x\}$, to effectively compute $B_1 \supset A_1$ and $B_2 \supset A_2$ with $B_1 \cap B_2 = \emptyset$ and $\langle B_1 \cup B_2 \rangle = V_\infty$. Indeed, if we let $\{y_i : i \in \omega\}$ be a recursive basis of V_∞ , then we could take

$$B_1 = \langle K\ell(\vec{0}, a) \cup \{-ny_i, -nx : n > 0, i \in \omega\} \rangle$$

and

$$B_2 = \langle ny_i, nx : n > 0, i \in \omega \rangle.$$

But if $D = H(a) \vee H(-a)$, then $D \leq B_1$. However $x \in B_1 - D$, giving the desired contradiction. $\square$

We do not know if $C(V_\infty)$ is a lower semilattice, but we conjecture that it is. The decidability (or lack thereof) of $\text{Th}(C(V_\infty))$ is also unknown. A relevant result [1] is that the theory of the lower semilattice of decidable (affine) subspaces is undecidable (although, of course, the lattice of recursive sets has a decidable theory).

We end this section with three simple results which show that many of the results of Sections 2 and 3 simply fail for $K(V_\infty)$.

Theorem 4.6 *There are no simple substructures of $K(V_\infty)$.*

Proof. Suppose V is a simple substructure of $K(V_\infty)$. It is easy to see that if $V \in K(V_\infty)$ and, for each vector $\vec{v} \in V_\infty$ and $n \in \omega$,

$$V \cap \langle \{m\vec{v} : m \in \omega \ \& \ m \geq n\} \rangle \neq \emptyset,$$

then $V = V_\infty$. Thus if $V \neq^* V_\infty$ in $K(V_\infty)$, then there must be a vector $\vec{w}$ and a $q \in \omega$ such that $V \cap \langle \{m\vec{w} : m \in \omega \ \& \ m \geq q\} \rangle = \emptyset$. But $\langle \{m\vec{w} : m \in \omega \ \& \ m \geq q\} \rangle \neq^* \emptyset$ in $K(V_\infty)$. Thus there are no simple substructures in $K(V_\infty)$. $\square$

Theorem 4.7 *There exist maximal complemented elements of $K(V_\infty)$.*

Proof. Let $x_0, x_1, \dots$ be a basis for V_∞ . Then for each $n \in \omega$, let

$$M_n = \left\{ v \in V_\infty : v = \lambda_0 x_0 + \sum_{i>0} \lambda_i x_i \ \& \ \forall i (\lambda_i \in F) \ \& \ \lambda_0 \leq n \right\}.$$

It is easy to see that each $M_n \in K(V_\infty)$ and that

$$C_n = \{ \beta x_0 : \beta \in F \ \& \ \beta > n \}$$

is a complement for M_n in $K(V_\infty)$. That is, clearly $M_n \cap C_n = \emptyset$ and $\langle M_n \cup C_n \rangle = V_\infty$.

Note that for each $n > 0$,

$$M_n \subseteq \langle M_0 \cup \{(n+1)x_0\} \rangle = \langle M_n \cup \{(n+1)x_0\} \rangle.$$

It follows that $V \in K(V_\infty)$, $V \supseteq M_0$ and $V \neq^* M_0$, and so for all $n > 0$, there must exist an $x \in V - M_n$. But this means that for all $n > 0$, there exists an $x \in V$ of the form $x = \lambda_0 x_0 + \sum_{i>0} \lambda_i x_i$ where $\lambda_i \in F$ for all i and $\lambda_0 > n$. But then it is easy to see that since $V \in K(V_\infty)$, $V = V_\infty$. Thus M_0 is a maximal, in fact supermaximal, complemented element of $K(V_\infty)$. Here we say that $M \in K(V_\infty)$ is supermaximal if $M \neq^* V_\infty$, and for all $V \in K(V_\infty)$ such that $V \supseteq M$, either $V =^* M$ or $V = V_\infty$. $\square$

Another theorem which doesn't work here is Sacks' splitting.

Theorem 4.8 *There exists a $V \in K(V_\infty)$ such that $V \neq^* \emptyset$, V is not recursive, and V has no Friedberg splittings. That is, there are no $U, W \in K(V_\infty)$ such that U and V are not recursive, $U \cap V = \emptyset$, and $\langle U \cup W \rangle = V$.*

Proof. Let x be any nonzero vector in V_∞ . Let $\{q_0, q_1, \dots\}$ be an r.e. lower cut in the rationals Q such that $q_0 < q_1 < \dots$, where $\lim_i q_i = r$ where r is a nonrecursive real. Then consider the ray $R = \langle px : p \leq q_i \text{ for } i \in \omega \rangle$. Any splitting of R must give $R = R_1 \cup R_2$, with R_1 a ray and R_2 a segment. Moreover, since R_1 and R_2 are r.e., R_1 must be a lower recursive cut of R . Hence $R_1 \equiv_T \emptyset$ and $R_2 \equiv_T R$. Thus R cannot be Friedberg split and hence R certainly cannot be Sacks' split. $\square$

5 Conclusions and open questions

We note that there are many theorems in the lattice of r.e. substructures of an effective closure system which cannot be covered by the very general setup of Section 1. For example, in the Boolean algebra $\tilde{N}$ with special set S equal to the set of atoms of $\tilde{N}$, Remmel [79] proved that there exist simple and maximal substructures M such that $\text{supp}_S(M) = S$. Such substructures cannot be generated by a subset of S . Moreover, in the setting of effective Steinitz closure systems, there are a large number of existence theorems concerning structures with no extendible basis, i.e., substructures which do not have a basis which is extendible to a recursive basis for M . In that case, extra axioms were introduced to handle these existence theorems. That is, assume $\mathcal{M} = (M, \text{cl})$ is an effective Steinitz system. Then the following axioms,

which have appeared in the literature, guarantee the existence of various substructures of $\mathcal{M}$ with no extendible basis. Recall $\text{cl}_B(A) = \text{cl}(A \cup B)$ for $A, B \subseteq M$.

Axiom I. Let V be closed and let I be an infinite independent set in (M, cl_V) . Then there exists a $z \in M$ such that in (M, cl_V) , $\text{supp}_I(z)$ has at least two members.

Axiom II. Let V be closed and let J be an infinite independent set in (V, cl) . Then the dimension of $\text{cl}(J \cup \{x\}) - \text{cl}(J)$ in (V, cl) is infinite.

Axiom III. There exists a $k > 0$ such that, for any infinite dimensional closed set I and independent set J in (M, cl_I) with $|J| > k$, we have the following:

For all $y \in J$, and all $F \subseteq J$ with $y \in F$ and $|F| = k$, and any $v_0, \dots, v_n$ outside of $\text{cl}_I(\emptyset)$, there exists an $x \in \text{cl}_I(F)$ with $v_0, \dots, v_n$ not in $\text{cl}_I(\{x\})$, such that $\text{supp}_F(x)$ in (M, cl_I) has at least two elements, including y .

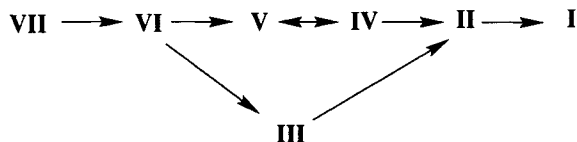
Axiom IV. (Downey's semiregularity.) No finite dimensional closed set is the union of two proper closed sets.

Axiom V_A . (Baldwin's federation over A .) There is a finite set A , such that for any finite independent set B in (M, cl) , there exists an x in $\text{cl}_A(B)$ which is not in any $\text{cl}_A(B')$ for any proper subset B' of B .

Axiom VI. (Baldwin's weak regularity.) No k -dimensional closed set is the union of k ($k - 1$)-dimensional closed subsets.

Axiom VII. (Regularity of Metakides and Nerode.) No finite dimensional closed set is a finite union of proper closed subsets.

The first three axioms are due to Nerode and Rummel [69, 71]. Indeed Nerode and Rummel [69, 71] completely classified the relations between these axioms. These relations may be summarized by the following diagram where the lack of an arrow means that the implication does not hold.



Moreover Nerode and Remmel [69, 71] proved that Axiom I implies the existence of maximal substructures with no extendible basis. Axiom II implies the existence of supermaximal substructures and the existence of effectively nowhere simple substructures with no extendible basis. Here a substructure $U \in L(\mathcal{M})$ is *supermaximal* if $U \neq^* M$, and for any r.e. substructure $W \supseteq U$, either $W =^* U$ or $W = M$. Axiom III implies the existence of a recursive supermaximal substructure and the existence of a recursive effectively nowhere simple substructure with no extendible basis. Axiom IV implies the existence of a V such that $D_i(V)$ is many-one incomparable to $D_j(V)$ for $i \neq j$. Here $D_j(V)$ is the j -th dependence set of V , i.e.,

$$D_j(V) = \{[v_1, \dots, v_j] : v_1, \dots, v_j \text{ are dependent over } V\}$$

where $[\ , \]$ is some effective pairing function. Finally Axiom $V_\emptyset$ implies the existence of supermaximal spaces with a prespecified sequence of dependence degrees, see [71].

We close by listing some open questions highlighted by our results. In some cases we choose to give specific instances of questions, rather than giving them in the general setting.

Question 5.1 (Remmel, Retzlaff, Guichard) Do r -maximal subsets of recursive bases of V_∞ generate r -maximal subspaces? The analogous question is open for most algebras. We remark that perhaps the techniques of Madan and Robinson [55] might be relevant if the answer is negative.

Question 5.2 (Remmel) Do atomless r.e. subsets of recursive bases generate atomless subspaces? The results of Downey [26] seem to indicate a negative answer. Again all analogous questions are open.

Question 5.3 What are the principal filters of $L^*(V_\infty)$? The usual arguments show that $L^*(V, \uparrow)$ must be a Σ_3^0 modular lattice. Remmel [77] modified Lachlan's $L(\omega)$ argument to embed every Σ_3^0 Boolean algebra as a filter of $L^*(V_\infty)$, and the general setting Steinitz closure system as a similar result may be found in [24]. Nerode and Smith [72] showed that every finite distributive lattice is a filter of $L(V_\infty)$. Downey [unpublished] proved that many finite modular lattices, plus some isolated infinite lattices (for example, "1- ∞ -1") which are not Σ_3^0 Boolean algebras, can be realized as filters of $L^*(V_\infty)$. We conjecture that every Σ_3^0 bounded modular lattice is a filter.

The situation for other algebras is completely open, apart from the coding result of [24] mentioned in Section 2.

Question 5.4 A related question here (stated for $L(V_\infty)$) suggested by our results is: Let A be an hh-simple subset of a recursive basis B , is $L^*(A, \uparrow)$ in $L(B)$ isomorphic to $L^*((A)^*, \uparrow)$ in $L(V_\infty)$? That is, is the lattice of supersets of A modulo finite sets isomorphic to the lattice of superspaces of $(A)^*$ modulo finite dimensional spaces?

Question 5.5 Are the $\vec{A} \vec{E}$ theories of, say, Steinitz algebras, always decidable? For $L(F_\infty)$, it is not known whether or not every $V \in L(F_\infty)$ has a major subfield, so one will need a lot of work to carry out a procedure similar to that of Lachlan's [54] in $L(F_\infty)$.

Question 5.6 In view of Ash and Rosenthal's [2] results on intersections of algebraically closed fields, perhaps some of the embedding questions might be more reasonably attacked. Understanding $L(F_\infty)$ would seem to be crucial to our understanding of the general Steinitz setting. In fact, Nerode has asked whether or not every Steinitz system is an intersection subsystem of (F_∞, cl) . A related question is whether or not every automorphism of $S(F_\infty)$ is induced by an automorphism of F_∞ . Analogous results hold in $S(V_\infty)$, this being the "fundamental theorem of projective geometry", and in $S(\tilde{Q})$, this being a result of Sachs [87]. These last two questions are classical ones.

Question 5.7 (Downey) Is $L^*(F_\infty)$ a lattice? Nerode and Remmel [69] observed that $=^*$ is not a lattice congruence on $L(F_\infty)$. For many other systems this question is also open.

Question 5.8 Do any of these algebras have interesting orbits? For example, Soare's famous theorem on maximal sets [92] shows that maximal sets form an orbit in the automorphism groups of $L^*(\omega)$ and $L(\omega)$.

Kalantari and Retzlaff [49] showed that this fails in $L(V_\infty)$, and Guichard [40] improved this to show that supermaximal subspaces also don't form an orbit.

This question is unanswered for $L^*(V_\infty)$. Downey asked similar questions for a type of maximal ideal in $LI(\tilde{Q})$ (see [25]) for which it seems conceivable that the automorphism machinery might apply (since the lattice is distributive). Downey, Jockusch, and Stob [27], combined with Cholak, Coles, and Downey, have proven that in this setting, the relevant notion of maximality does indeed define an invariant class of degrees, namely the *anr* degrees, so that a version of Martin's Theorem holds. In other settings the situation is completely open.

References

- [1] C. J. Ash and R. G. Downey, Decidable subspaces and recursively enumerable subspaces, *J. Symbolic Logic*, **49** (1984) 1137–1145.
- [2] C. J. Ash and J. W. Rosenthal, Intersections of algebraically closed fields, *Ann. Pure Appl. Logic*, **30** (1986) 103–119.
- [3] J. T. Baldwin, Recursion theory and abstract dependence, in: *Patras Logic Symposium*, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) G. Metakides (ed.), *Stud. Logic Found. Math.*, **109** (1982), 67–76.
- [4] J. T. Baldwin, First-order theories of abstract dependence relations, *Ann. Pure Appl. Logic*, **26** (1984) 215–243.
- [5] F. A. Bäuerle, Complexity Theoretic Algebra, Ph.D. Thesis, Univ. California, San Diego, La Jolla, CA, (1994).
- [6] F. A. Bäuerle and J. B. Remmel, On speedable and levelable vector spaces, *Ann. Pure Appl. Logic*, **67** (1994) 61–112.
- [7] G. Baumslag, G. E. Dyer and C. Miller III, On the integral homology of finitely presented groups, *Topology*, **22** (1983) 27–46.
- [8] V. L. Bennison, Information content characterizations of complexity-theoretic properties, in: *Theoretical Computer Science*, (4th. GI Conf., Aachen, 1979), *Lecture Notes in Comput. Sci.*, **67** (1979) 58–66.
- [9] M. Blum, A machine independent theory of the complexity of recursive functions, *J. Assoc. Comput. Mach.*, **14** (1967) 322–336.
- [10] M. Blum and I. Marques, On complexity properties of recursively enumerable sets, *J. Symbolic Logic*, **38** (1973) 579–593.
- [11] J. S. Carroll, Recursively enumerable equivalence relations, Ph.D. Thesis, Univ. Wisconsin, Madison, WI, (1983).
- [12] J. S. Carroll, Some undecidability results for lattices in recursion theory, *Pacific J. Math.*, **122** (1986) 319–331.
- [13] D. Cenzer and J. B. Remmel, Π_1^0 classes in mathematics, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 623–821.

- [14] P. A. Cholak, R. Coles and R. G. Downey, Invariance, degrees and Π_1^0 classes, (to appear).
- [15] J. N. Crossley, (ed.), Aspects of Effective Algebra, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981).
- [16] J. N. Crossley and A. Nerode, Combinatorial Functors, *Ergeb. Math. Grenzgeb. Ser. 3*, **81** (1974).
- [17] J. C. E. Dekker, Countable vector spaces with recursive operations. Part I, *J. Symbolic Logic*, **34** (1969) 363–387.
- [18] J. C. E. Dekker, Countable vector spaces with recursive operations. Part II, *J. Symbolic Logic*, **36** (1971) 477–493.
- [19] R. G. Downey, Nowhere simplicity in matroids, *J. Austral. Math. Soc., Ser. A*, **35** (1983) 28–45.
- [20] R. G. Downey, On a question of A. Retzlaff, *Z. Math. Logik Grundlag. Math.*, **29** (1983) 379–384.
- [21] R. G. Downey, Co-immune subspaces and complementation in V_∞ , *J. Symbolic Logic*, **49** (1984) 528–538.
- [22] R. G. Downey, Some remarks on a theorem of Iraj Kalantari concerning convexity and recursion theory, *Z. Math. Logik Grundlag. Math.*, **30** (1984) 295–302.
- [23] R. G. Downey, Sound, totally sound, and unsound recursive equivalence types, *Ann. Pure Appl. Logic*, **31** (1986) 1–22.
- [24] R. G. Downey, Undecidability of $L(F_\infty)$ and other lattices of r.e. substructures, *Ann. Pure Appl. Logic*, **32** (1986) 17–26; corr. *ibid.*, **48** (1990) 299–301.
- [25] R. G. Downey, Maximal theories, *Ann. Pure Appl. Logic*, **33** (1987) 245–282.
- [26] R. G. Downey, Orbits of creative subspaces, *Proc. Amer. Math. Soc.*, **99** (1987) 163–170.

- [27] R. G. Downey, C. G. Jockusch Jr. and M. Stob, Array nonrecursive sets and multiple permitting arguments, in: *Recursion Theory Week*, (Proc. Conf., Oberwolfach, Mar. 19–25, 1989), K. Ambos-Spies G. H. Müller and G. E. Sachs, (eds.), *Lecture Notes in Math.*, **1432** (1990) 141–173.
- [28] R. G. Downey and I. Kalantari, Effective extensions of linear forms on a recursive vector space over a recursive field, *Z. Math. Logik Grundlag. Math.*, **31** (1985) 193–200.
- [29] R. G. Downey and J. B. Remmel, Automorphisms and recursive structures, *Z. Math. Logik Grundlag. Math.*, **33** (1987) 339–345.
- [30] R. G. Downey and J. B. Remmel, Classification of degree classes associated with r.e. subspaces, *Ann. Pure Appl. Logic*, **42** (1989) 105–124.
- [31] R. G. Downey and J. B. Remmel, Effectively and noneffectively nowhere simple subspaces, in: *Logical Methods*, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), *Progr. Comput. Sci. Appl. Logic*, **12** (1993) 314–351.
- [32] R. G. Downey, J. B. Remmel and L. V. Welch, Degrees of splittings and bases of recursively enumerable subspaces, *Trans. Amer. Math. Soc.*, **302** (1987) 683–714.
- [33] R. G. Downey and M. Stob, Structural interactions of the recursively enumerable T - and W -degrees, (Special issue: 2nd. Southeast Asian Logic Conf., Bangkok, 1984), *Ann. Pure Appl. Logic*, **31** (1986) 205–236.
- [34] L. J. Feiner, Hierarchies of Boolean algebras, *J. Symbolic Logic*, **35** (1970) 365–374.
- [35] I. S. Filotti, On effectively levelable sets, *Recursive Function Theory Newsletter*, **2** (1972) 12–13.
- [36] A. Fröhlich and J. C. Shepherdson, Effective procedures in field theory, *Philos. Trans. Roy. Soc. London, Ser. A*, **248** (1956) 407–432.
- [37] R. Guhl, Two Types of Recursively Enumerable Vector Spaces, Ph.D. Thesis, Rutgers Univ., New Brunswick, NJ, (1973).
- [38] J. T. Gill III and P. H. Morris, On subcreative sets and S -reducibility, *J. Symbolic Logic*, **39** (1974) 669–677.

- [39] D. Guichard, Automorphisms of substructure lattices in recursive algebra, *Ann. Pure Appl. Logic*, **25** (1983) 47–58.
- [40] D. Guichard, A note on r -maximal subspaces of V_∞ , *Ann. Pure Appl. Logic*, **26** (1984) 1–9.
- [41] L. Harrington, The undecidability of the lattice of recursively enumerable sets, (handwritten notes, 1982).
- [42] E. Hermann, The undecidability of the elementary theory of the lattice of recursively enumerable sets, in: *Frege Conf.*, (Proc. Internatl. Conf., Schwerin, Sept. 10–14, 1984), G. Wechsung, (ed.), *Math. Res.*, **20** (1984) 66–72.
- [43] E. Hermann, Orbits of Σ_3^0 sets and the lattice of recursively enumerable sets, (to appear).
- [44] A. G. Higman, Subgroups of finitely presented groups, *Proc. Roy. Soc. London Ser. A*, **262** (1961) 455–475.
- [45] I. Kalantari, Major subspaces of recursively enumerable vector spaces, *J. Symbolic Logic*, **43** (1978) 293–303.
- [46] I. Kalantari, Effective content of a theorem of M. H. Stone, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981) 128–146.
- [47] I. Kalantari, Major subsets in effective topology, in: *Patras Logic Symposium*, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) G. Metakides (ed.), *Stud. Logic Found. Math.*, **109** (1982) 77–94.
- [48] I. Kalantari, The Hahn-Banach theorem and its effectiveness, (unpublished manuscript).
- [49] I. Kalantari and A. Retzlaff, Maximal vector spaces under automorphisms of the lattice of recursively enumerable vector spaces, *J. Symbolic Logic*, **42** (1977) 481–491.
- [50] I. Kalantari and A. Retzlaff, Recursive constructions in topological spaces, *J. Symbolic Logic*, **44** (1979) 609–625.

- [51] G. Kothe, *Topologische lineare Räume I*, Grundlehren Math. Wiss., **107** (1st. edn. 1960, 2nd. edn. 1966); [translated as: *Topological Vector Spaces I*, translated by D. J. H. Garling, Grundlehren Math. Wiss., **159** (1969)].
- [52] G. Kothe, *Topological Vector Spaces II*, Grundlehren Math. Wiss., **237** (1979)].
- [53] A. H. Lachlan, On the lattice of recursively enumerable sets, *Trans. Amer. Math. Soc.*, **130** (1968) 1–37.
- [54] A. H. Lachlan, The elementary theory of recursively enumerable sets, *Duke Math. J.*, **35** (1968) 123–146.
- [55] P. B. Madan and R. W. Robinson, Monotone and 1-1 sets, *J. Austral. Math. Soc., Ser. A*, **33** (1982) 62–75.
- [56] W. Magnus, A. Karass and D. Solitar, *Combinational group theory*, (1st. edn., Wiley, NY, 1966; 2nd. rev. edn., Dover, NY, 1976).
- [57] D. A. Martin, Classes of recursively enumerable sets and degrees of unsolvability, *Z. Math. Logik Grundlag. Math.*, **12** (1966) 295–310.
- [58] D. A. Martin and M. B. Pour-El, Axiomatizable theories with few axiomatizable extensions, *J. Symbolic Logic*, **35** (1970) 205–209.
- [59] Yu. V. Matijasevič, Enumerable sets are diophantine (Russian), *Dokl. Akad. Nauk SSSR*, **191** (1970) 279–282; [translated in: *Soviet Math. – Dokl.*, **11** (1970) 354–357].
- [60] Yu. V. Matijasevič, Diophantine representation of enumerable predicates (Russian), *Izv. Akad. Nauk SSSR, Ser. Mat.*, **35** (1971) 3–30; [translated in: *Math. USSR–Izvestija*, **5** (1971) 1–28].
- [61] G. Metakides, (ed.), *Patras Logic Symposion*, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) *Stud. Logic Found. Math.*, **109** (1982).
- [62] G. Metakides and A. Nerode, Recursion theory and algebra, in: *Algebra and Logic*, (Proc. 14th. Summer Res. Inst. Austral. Math. Soc., Monash Univ., Clayton, Vic., Australia, Jan. 6 – Feb. 16, 1974), J. N. Crossley, (ed.), *Lecture Notes in Math.*, **450** (1975) 209–219.
- [63] G. Metakides and A. Nerode, Recursively enumerable vector spaces, *Ann. Math. Logic*, **11** (1977) 147–171.

- [64] G. Metakides and A. Nerode, Effective content of field theory, *Ann. Math. Logic*, **17** (1979) 289–320.
- [65] G. Metakides and A. Nerode, Recursion theory on fields and abstract dependence, *J. Algebra*, **65** (1980) 36–59.
- [66] G. Metakides and J. B. Remmel, Recursion theory on orderings I, A model theoretic setting, *J. Symbolic Logic*, **44** (1979) 383–402.
- [67] D. Miller and J. B. Remmel, Effectively nowhere simple sets, *J. Symbolic Logic*, **49** (1984) 129–136.
- [68] J. Myhill, Creative sets, *Z. Math. Logik Grundlag. Math.*, **1** (1955) 97–108.
- [69] A. Nerode and J. B. Remmel, Recursion theory on matroids. in: *Patras Logic Symposion*, (Proc. Logic Sympos. Patras, Greece, Aug. 18–22, 1980) G. Metakides (ed.), *Stud. Logic Found. Math.*, **109** (1982) 41–65.
- [70] A. Nerode and J. B. Remmel, A survey of lattices of r.e. substructures, in: *Recursion Theory*, (Proc. AMS-ASL Summer Inst., Ithaca, NY, June 28 – July 16, 1982), A. Nerode and R. A. Shore, (eds.), *Proc. Sympos. Pure Math.*, **42** (1985) 323–375.
- [71] A. Nerode and J. B. Remmel, Generic objects in recursion theory II, (Special issue: 2nd. Southeast Asian Logic Conf., Bangkok, 1984), *Ann. Pure Appl. Logic*, **31** (1986) 257–288.
- [72] A. Nerode and R. L. Smith, The undecidability of the lattice of recursively enumerable subspaces, in: *Proc. 3rd. Brazilian Conf. on Math. Logic*, (Inst. Math., Fed. Univ. Pernambuco, Recife, 1979), A. I. Arruda, N. C. A. da Costa and A. M. Sette, (eds.), *Soc. Brasil Logica*, Sao Paulo, (1980) 245–252.
- [73] T. A. Nevins, Degrees of convex dependence in recursively enumerable vector spaces, *Ann. Pure Appl. Logic*, **60** (1993) 31–47.
- [74] M. B. Pour-El and J. I. Richards, Noncomputability in analysis and physics: a complete determination of the class of noncomputable linear operators, *Adv. Math.*, **48** (1983), 44–74.
- [75] M. O. Rabin, Computable algebra, general theory and theory of computable fields, *Trans. Amer. Math. Soc.*, **95** (1960) 341–360.

- [76] J. B. Remmel, Maximal and cohesive vector spaces, *J. Symbolic Logic*, **42** (1977) 400–418.
- [77] J. B. Remmel, On the lattice of r.e. superspaces of a r.e. space, *Notices Amer. Math. Soc.*, **24** (1977) A 299 (abstract 77T–E26).
- [78] J. B. Remmel, An r -maximal vector space not contained in any maximal vector space, *J. Symbolic Logic*, **43** (1978) 430–441.
- [79] J. B. Remmel, Recursively enumerable Boolean algebras, *Ann. Math. Logic*, **15** (1978) 75–107.
- [80] J. B. Remmel, r -maximal Boolean algebras, *J. Symbolic Logic*, **44** (1979) 533–548.
- [81] J. B. Remmel, On r.e. and co-r.e. vector spaces with nonextendible bases, *J. Symbolic Logic*, **45** (1980) 20–34.
- [82] J. B. Remmel, Recursion theory on algebraic structures with independent sets, *Ann. Math. Logic*, **18** (1980) 153–191.
- [83] A. Retzlaff, Simple and hyperhypersimple vector spaces, *J. Symbolic Logic*, **43** (1978) 260–269.
- [84] A. Retzlaff, Direct summands of recursively enumerable vector spaces, *Z. Math. Logik Grundlag. Math.*, **25** (1979) 363–372.
- [85] F. Richman, The constructive theory of countable abelian p -groups, *Pacific J. Math.*, **45** (1973) 621–637.
- [86] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, (1st. edn., McGraw-Hill, New York-Toronto, Ont.-London, 1967; 2nd. edn., MIT Press, Cambridge, Mass., London, 1987).
- [87] D. Sachs, The lattice of subalgebras of a Boolean algebra, *Canad. J. Math.*, **14** (1962) 451–460.
- [88] G. E. Sacks, On the degrees less than $0'$, *Ann. of Math.*, **77** (1963) 211–231.
- [89] R. A. Shore, Controlling the dependence degree of a recursively enumerable vector space, *J. Symbolic Logic*, **43** (1978) 13–22.
- [90] R. A. Shore, Nowhere simple sets and the lattice of recursively enumerable sets, *J. Symbolic Logic*, **43** (1978) 322–330.

- [91] R. I. Soare, Recursion Theory and Dedekind Cuts, Ph.D. Thesis, Cornell Univ., Ithaca, NY, (1967).
- [92] R. I. Soare, Automorphisms of the lattice of recursively enumerable sets, part I: maximal sets, *Ann. of Math.*, **100** (1974) 80–120.
- [93] R. I. Soare, The infinite injury priority method, *J. Symbolic Logic*, **41** (1976) 513–530.
- [94] R. I. Soare, Computational complexity, speedable and levelable sets, *J. Symbolic Logic*, **42** (1977) 545–563.
- [95] R. I. Soare, Recursively enumerable sets and degrees, *Bull. Amer. Math. Soc. (N.S.)*, **84** (1978) 1149–1181.
- [96] R. I. Soare, Recursively Enumerable Sets and Degrees. A Study of Computable Functions and Computably Generated Sets, *Perspect. in Math. Logic*, (1987).
- [97] J. Yang, On recursive Boolean algebras, Ph.D. Thesis, Univ. California, San Diego, La Jolla, CA, (1996).

R. G. Downey
Department of Mathematics
University of Victoria
PO Box 600
Wellington
New Zealand
e-mail: 1rod.downey@vuw.ac.nz1

J. B. Remmel
Department of Mathematics
University of California at San Diego
La Jolla, CA 92093, USA
email: jremmel@ucsd.edu

This Page Intentionally Left Blank

Chapter 16

A Survey of Recursive Combinatorics

W. Gasarch*

Contents

- Introduction
- 1. Definitions and notation
- 2. König's Lemma
- 3. Ramsey's Theorem
- 4. Coloring infinite graphs
- 5. Hall's Theorem on bipartite graphs
- 6. Dilworth's Theorem for partial orders
- 7. Miscellaneous results in recursive combinatorics
- References

Introduction

Many theorems in infinite combinatorics have noneffective proofs. Nerode's recursive mathematics program [124] involves looking at noneffective proofs and seeing if they can be made effective. The framework is recursion-theoretic. For example, to see if the theorem (which we denote T) "Every vector space has a basis" has an effective proof, one might look at the statement (which we denote S) "Given a recursive vector space, one can effectively find a basis for it". If statement S is false, then there can be no effective proof of Theorem T (statement S is false, see [123]). We examine theorems about infinite combinatorics in this context. Given a theorem in infinite combinatorics that has a noneffective proof, we ask the following three questions:

*Supported in part by NSF grants CCR-84-05079, CCR-88-03641, and CCR-90-20079.

- (1) Is the recursive analogue true? (Usually no.)
- (2) Is some modification of the recursive analogue true? (Usually yes.) The modification can be either
 - (a) recursion-theoretic (i.e., in Theorem T above we might replace “one can effectively find” with “one can find recursively in the oracle A ”, for some well behaved A), or
 - (b) combinatorial (i.e., change the type of object you want to find).
- (3) How hard is it (in the arithmetic hierarchy) to determine if a given instance of the theorem has a recursive solution?

Item (1) is an example of recursive mathematics. This field has its roots in two early papers in recursive algebra [56, 136]; however, Nerode is the modern founder of recursive mathematics [124]. Item (2)(a) is an attempt to measure just how noneffective the proofs are, and is evident in the work of Jockusch on Ramsey’s Theorem [87]. Item (2)(b) is an attempt to recover the effective aspects of combinatorics in infinite domains, and was first mentioned by Kierstead in his work on Dilworth’s theorem for infinite partial orders [94]. Item (3) was an outgrowth of an attempt to link recursive graph theory to complexity theory. For example, 3-colorability of finite graphs is of unknown complexity (since it is NP-complete), so the problem of determining if an infinite graph is 3-colorable might be a good analogue. This was the (unstated) motivation behind the first paper that analyzed such issues [14]. Item (3) was first pursued by Beigel and Gasarch [14]. Subsequent work has been done by Harel [76], and Gasarch and Martin [66].

There are not many published papers pursuing item (3), so many such results appear here for the first time.

General philosophy

In each section of this paper we will state a noneffective theorem, sketch a proof, and then consider possible recursive analogues and their modifications. More detail than usual will be given in the proofs of the noneffective theorems. This is because (a), if we want to examine a noneffective proof then that proof ought to be in this paper, and (b), these proofs tend to not get written down, as most authors (rightly so) just say “by the usual compactness arguments”, or “by König’s lemma on infinite trees”.

Some of the proofs in this paper look similar and can probably be put into an abstract framework. Indeed (a), abstract frameworks for constructions of recursive partial orders [97] and recursive graphs [32] have been worked out, and (b), an abstract framework for recursion-theoretic theorems, namely the theory of Π_1^0 classes [36] has been worked out. We do not use these or other frameworks, because such devices make reading more difficult for readers not familiar with the area.

Each section has a subsection of miscellaneous results, as does the paper. The results mentioned here are not proven, and are intended more to point the reader to references.

It was my intention to mention every single result in recursive combinatorics that was known. While it is doubtful that I've succeeded, I believe I have come close.

Related work

Downey has written a survey [49] of recursive linear orderings. In addition, the last chapter of Rosenstein's book [143] is on recursive linear orderings. Cenzer and Remmel have written a survey [36] on Π_1^0 classes. These arise often in recursive combinatorics and in other parts of recursive mathematics.

In this survey we often give index-set results about how hard certain combinatorial results are. Cenzer and Remmel [37] have refined some of those results.

1 Definitions and notation

In this section we present definitions and notations that are used throughout this paper. We do not define notions relevant to combinatorial objects in this section, but rather in the section where they are used (e.g., "homogeneous set" is defined in the section on Ramsey Theory).

All recursion-theoretic notation is standard, and follows [159].

Notation 1.1 Let $A \subseteq \mathbb{N}$ and $k \in \mathbb{N}$.

- (1) $[A]^k$ is the set of all k -element subsets of A .
- (2) $\exists^\infty$ means "for infinitely many".

- (3) $\forall^\infty$ means “for all but a finite number of”.
- (4) $\mu x[P(x)]$ means “the least x such that $P(x)$ holds”.
- (5) $A(\)$ and χ_A both represent the characteristic function of A .
- (6) $\langle x, y \rangle$ is a recursive bijection from $\mathbb{N} \times \mathbb{N}$ to $\mathbb{N}$. Similarly for $\langle x_1, \dots, x_n \rangle$. All functions in this paper are on one variable; however we may write $f(\langle x, y \rangle)$ and $\{e\}(\langle x, y \rangle)$ as $f(x, y)$ and $\{e\}(x, y)$.
- (7) $\{0\}, \{1\}, \dots$ is a standard list of all Turing machines. We use $\{e\}$ for both the Turing machine and the partial recursive function that it computes. W_e is the domain of $\{e\}$.
- (8) $\{0\}^{(\)}, \{1\}^{(\)}, \dots$ is a standard list of all oracle Turing machines. We use $\{e\}^A$ for both the oracle Turing machine with oracle A , and the partial recursive-in- A function that it computes. W_e^A is the domain of $\{e\}^A$.
- (9) K is the halting set, K_s denotes the first s elements in some fixed recursive enumeration of K . TOT01 denotes the set of indices of total functions that are 0–1 valued. TOTINF01 denotes the set of indices of total functions that are 0–1 valued and take on the value 0 infinitely often. TOT $_a$ is the set of indices of total functions whose image is contained in $\{1, \dots, a\}$.
- (10) If A is a set, then A' , the *jump of A* , is $\{e : \{e\}^A(e) \downarrow\}$, i.e., the halting set relative to A . If f is a function, then f' , the *jump of f* , is $\{e : \{e\}^f(e) \downarrow\}$, where $\{e\}^{(\)}$ is defined in such a way as to be able to access a function instead of a set.
- (11) If σ is a finite sequence of natural numbers, then we think of it as being a function from $\{0, 1, \dots, |\sigma| - 1\}$ to $\mathbb{N}$, which we also denote by σ . The value of $\sigma(i)$ is the $(i + 1)$ -st element of the sequence (we do this since there is no “0-th element of a sequence”). Hence the value of $\sigma(i)$ is the value of the function at i .
- (12) If σ is an infinite sequence of natural numbers, then we think of it as being a function from $\mathbb{N}$ to $\mathbb{N}$, and the same conventions as in the last item apply.

- (13) If σ is a finite sequence and τ is a finite or infinite sequence, then $\sigma \preceq \tau$ means that σ is a prefix of τ , and $\sigma \prec \tau$ means that σ is a proper prefix of τ .
- (14) If Σ is a set, then Σ^* is the set of finite strings over Σ . The most usual uses are $\{0, 1\}^*$ and $\mathbb{N}^*$.
- (15) λ denotes the empty string.

In this paper we often deal with functions where we care about the answer only if the input is of a certain form. We will ignore what happens otherwise. For example, in Section 4 we will have a convention by which certain numbers represent graphs, and others do not; and we may want $f(e)$ to be meaningful only if e represents a graph. For this reason we introduce the notion of a *Promise Problem*, originally defined (in a complexity theory context) in [53].

Definition 1.2 A *promise problem* is a set D , together with a partial function f , such that D is a subset of the domain of f . Let (D, f) be a promise problem, and X be a set. A *solution* to (D, f) is a total function g that agrees with f on D . (D, f) is *recursive in* X if there is a solution g to (D, f) such that $g \leq_T X$. X is *recursive in* (D, f) if, for every solution g to (D, f) , $X \leq_T g$.

Definition 1.3 Let (D, A) be a promise problem where A is a 0–1 valued partial function. A *solution* to (D, A) is a set B that agrees with A on D . (D, A) is Σ_n if some solution is a Σ_n set. (D, A) is Σ_n -hard if all solutions are Σ_n -hard sets. A promise problem is Σ_n -complete if it is both in Σ_n and is Σ_n -hard. The same definitions apply to Π_n .

2 König's Lemma

König's Lemma is an important theorem in infinite combinatorics. Many theorems in infinite combinatorics can be derived from it, including Theorems 3.3, 4.3, 5.5, and 6.3. We do not prove these theorems using König's lemma, since we want each chapter to be self contained.

In this section we present a proof of the classical König's Lemma, then show that a recursive analogue is false and give an index set version, and then state (but do not prove) a recursion-theoretic analogue that is true.

The latter is due to Jockusch and Soare [90], and is called the “low basis theorem”. We will use it in later chapters to obtain recursion-theoretic versions of theorems in infinite combinatorics.

2.1 Definition and classical version

Definition 2.1 A *tree* is a subset T of $\mathbb{N}^*$ such that if $\sigma \in T$ and $\tau \preceq \sigma$, then $\tau \in T$. A tree is *bounded* if there exists a function g such that, for all $\sigma \in T$, $|\sigma| \geq n + 1 \Rightarrow \sigma(n) \leq g(n)$. (Recall that $\sigma(n)$ is the $(n + 1)$ -st element of the sequence σ .)

Definition 2.2 Let T be a tree. An *infinite branch* of T is an infinite sequence σ , such that every finite prefix of σ is in T . To each infinite branch σ , we associate a function as indicated in Notation 2.1 (12). We view T as a set of functions by identifying T with the set of infinite branches of T .

Theorem 2.3 (König’s Lemma [106]) *If T is an infinite tree and T is bounded, then T has an infinite branch.*

Proof. Let $\sigma_0 = \lambda$ and note that $\sigma_0 \in T$. Assume inductively that $\sigma_n \in T$ and the set $\{\sigma \in T : \sigma_n \preceq \sigma\}$ is infinite. Let

$$a_n = \mu x [\sigma_n x \in T \wedge |\{\sigma \in T : \sigma_n x \preceq \sigma\}| = \aleph_0]$$

$$\sigma_{n+1} = \sigma_n x$$

For all n , a_n exists because T is bounded. The sequence $a_0, a_1, \dots$ is an infinite branch of T . □

The proof of Theorem 2.3 given above is noneffective. To see if the proof could have been made effective we will look at a potential analogue. In order to state this analogue we need some definitions.

Definition 2.4 A tree T is *recursive* if the set T is recursive. A tree T is *recursively bounded* if T is recursive, T is bounded, and the bounding function is recursive.

Definition 2.5 A number $e = \langle e_1, e_2 \rangle$ is an *index* for a recursively bounded tree if

- (1) $e_1 \in \text{TOT01}$ and decides a set we denote T ,
- (2) the set T is a tree,
- (3) $e_2 \in \text{TOT}$ and $\{e_2\}$ is a bounding function for T .

T is the tree determined by e . The tree determined by e is denoted T_e . We denote the set of indices for recursively bounded trees by TREE . Note that TREE is Π_2 .

Potential Analogue 2.6 There is a recursive algorithm $\mathcal{A}$ that performs the following. Given an index e for an infinite recursively bounded tree T , $\mathcal{A}$ outputs an index for a recursive infinite branch. A consequence is that all infinite recursively bounded trees have recursive infinite branches.

2.2 Recursive analogue is false

We show that the potential analogue is false. This was first shown by Kleene [104] and seems to be the first negative result in recursive combinatorics.

Definition 2.7 A pair of sets A and B is *recursively inseparable* if $A \cap B = \emptyset$, and there is no recursive R such that $A \subseteq R$ and $B \subseteq \overline{R}$.

Note 2.8 The sets $A = \{x : \varphi_x(x) \downarrow = 0\}$ and $B = \{x : \varphi_x(x) \downarrow = 1\}$ are easily seen to be a pair of r.e. recursively inseparable sets.

Theorem 2.9 There exists an infinite recursively bounded tree with no infinite recursive branches.

Proof. Let A, B be a pair of r.e. sets that are recursively inseparable. We define a recursively bounded tree T such that every infinite branch of T codes a set that separates A and B . Let T be defined by $\sigma \in T$ iff

- (1) $(\forall i < |\sigma|) \sigma(i) \in \{0, 1\}$,
- (2) $(\forall i < |\sigma|) i \in A_{|\sigma|} \Rightarrow \sigma(i) = 1$, and
- (3) $(\forall i < |\sigma|) i \in B_{|\sigma|} \Rightarrow \sigma(i) = 0$.

Clearly T is recursively bounded, and every infinite branch of T is the characteristic function of a set that separates A and B . Since A and B are recursively inseparable, T has no infinite recursive branches. $\square$

Theorem 2.10 *The set*

$$\text{RECBRANCH} = \{e \mid T_e \text{ has a recursive infinite branch}\}.$$

is Σ_3 -complete.

Proof. RECBRANCH consists of all $\langle e_1, e_2 \rangle \in \text{TREE}$ such that there exists i with the following properties:

- (1) $i \in \text{TOT01}$.
- (2) $(\forall \sigma) [\{i\}(\sigma) = 1 \Rightarrow \{e_1\}(\sigma) = 1]$.
- (3) $(\forall \sigma)(\exists x)(\forall y \leq \{e_2\}(|\sigma| + 1), y \neq x)$

$$[\{i\}(\sigma) = 1 \Rightarrow (\{i\}(\sigma x) = 1 \wedge \{i\}(\sigma y) = 0)].$$

Clearly RECBRANCH is in Σ_3 .

To show that RECBRANCH is Σ_3 -complete, we use the Σ_3 -complete set

$$\text{SEP} = \{\langle x, y \rangle : W_x \text{ and } W_y \text{ are recursively separable}\}.$$

(For a proof that this set is Σ_3 -complete see [159].) We show

$$\text{SEP} \leq_m \text{RECBRANCH}.$$

Given $\langle x, y \rangle$, we define a recursively bounded tree T such that every infinite branch of T codes a set that separates W_x and W_y . Let T be defined by $\sigma \in T$ iff the following hold.

- (1) $(\forall i < |\sigma|) [\sigma(i) \in \{0, 1\}]$.
- (2) $(\forall i < |\sigma|) [i \in W_{x,|\sigma|} \Rightarrow \sigma(i) = 1]$.
- (3) $(\forall i < |\sigma|) [i \in W_{y,|\sigma|} \Rightarrow \sigma(i) = 0]$.

Clearly T is recursively bounded and every infinite branch of T is the characteristic function of a set that separates W_x and W_y . Hence T has an infinite recursive branch iff $\langle x, y \rangle \in \text{SEP}$. $\square$

2.3 Recursion-theoretic modifications

Kreisel [109] showed that every infinite recursively bounded tree has a branch $B \leq_T K$ (this follows easily from examining the proof of König's Lemma). Shoenfield [149] improved this to $B <_T K$. Jockusch and Soare [90] improved this further to $B' \leq_T K$. This is referred to as “the low basis theorem”.

We introduce some notation which we will use later when applying the low basis theorem, then state the theorem in that notation.

Definition 2.11 A set of functions $\mathcal{F}$ is Π_1^0 if there exists a recursive predicate R , such that

$$f \in \mathcal{F} \quad \text{iff} \quad (\forall n) [R(\langle f(0), \dots, f(n) \rangle)].$$

If, in addition, there is a recursive function g such that

$$(\forall f \in \mathcal{F})(\forall n) [f(n) \leq g(n)],$$

then $\mathcal{F}$ is called a *recursively bounded Π_1^0 class*. This definition easily relativizes to $\Pi_1^{0,A}$ classes by taking $R \leq_T A$.

Theorem 2.12 (Low Basis Theorem [90]) *If $\mathcal{F}$ is a nonempty recursively bounded Π_1^0 class, then there exists $f \in \mathcal{F}$ such that $f' \leq_T K$.*

There is also a relativized form.

Theorem 2.13 (Relativized Low Basis Theorem) *Let B be a set and $\mathcal{F}$ be a nonempty $\Pi_1^{0,B}$ class. If there exists $g \leq_T B$ such that*

$$(\forall f \in \mathcal{F})(\forall n \in \mathbb{N}) [f(n) \leq g(n)],$$

then there exists $f \in \mathcal{F}$ such that $f' \leq_T B'$.

To apply the theorem later, and to see that it yields a recursion-theoretic version of König's Lemma we need the following proposition.

Proposition 2.14 *If T is a recursive (recursively bounded) tree, then the set of infinite branches forms a (recursively bounded) Π_1^0 class.*

Corollary 2.15 *If T is an infinite recursively bounded tree, then T has an infinite branch B such that $B' \leq_T K$.*

2.4 Miscellaneous

Carstens and Golze [30] considered adding some number of cross connections to the tree, and seeing if it then had a recursive path. A *tree with cross connections* is a graph that can be looked at as a tree with some of the vertices on a level connected to each other. The n -th level is *saturated* if, for all sets of n vertices on that level, there exist two that are connected. A tree is *highly recursive* if, given a node, you can determine all of its children. Carstens and Golze showed that if G is a highly recursive tree with cross connections such that every level is saturated, then there exists a recursive infinite path. They were motivated by questions about one-dimensional cell spaces.

3 Ramsey's Theorem

We consider Ramsey's Theorem on colorings of $[N]^2$. We present the classical proof of Ramsey's Theorem on colorings of $[N]^2$ due to Ramsey [137] (see [68] for several proofs), then show that a recursive analogue of Ramsey's Theorem is false and give an index-set version, and then show that there are two recursion-theoretic modifications that are true. We will then state some results in proof theory that are related to this work, and finally state some miscellaneous results.

3.1 Definitions and classical version

Definition 3.1 A k -coloring of $[N]^m$ is a map from $[N]^m$ into $\{1, 2, \dots, k\}$. (It does not need to satisfy any additional properties.) The elements of $\{1, 2, \dots, k\}$ are called *colors*. If $k = 2$, then we may refer to 1 as 'RED' and 2 as 'BLUE' (note that RED < BLUE).

Definition 3.2 Let c be a k -coloring of $[N]^m$, i be a color, and $A \subseteq N$. A is *i -homogeneous with respect to c* if

- (1) A is infinite, and
- (2) for all distinct $x_1, \dots, x_m \in A$, $c(\{x_1, \dots, x_m\}) = i$.

A is *homogeneous with respect to c* if there is an i such that A is i -homogeneous with respect to c . We often drop the "with respect to c " if the coloring is clear from the context.

We exhibit a whimsical scenario to illustrate these concepts. Suppose that you host a party with a countably infinite number of guests. Assume their names are $0, 1, 2, \dots$. Color each pair of guests RED if they know each other, and BLUE if they do not. This is a 2-coloring of $[N]^2$. A RED-homogeneous set is an infinite set of people all of whom know each other, and a BLUE-homogeneous set is an infinite set of people no two of whom know each other.

We will later see that Ramsey's Theorem (infinite version) guarantees that there is either a RED-homogeneous set or a BLUE-homogeneous set. This was first proven by Ramsey [137]. For more on Ramsey theory (mostly finite versions) see [68].

We study the following simplified version of Ramsey's Theorem (the general version involves m -colorings of $[N]^k$). We give a direct proof; it can also be proven by König's Lemma (Theorem 2.3).

Theorem 3.3 *If c is a k -coloring of $[N]^2$, then there exists a homogeneous set.*

Proof. The variable d will range over the colors $\{1, \dots, k\}$.

Let $A_0 = N$ and $a_1 = 1$. Assume inductively that $A_{n-1} \subseteq N$, $a_1, \dots, a_n \in N$, and $c_1, \dots, c_{n-1} \in \{1, \dots, k\}$ have been defined such that

- (1) A_{n-1} is infinite,
- (2) $a_1, \dots, a_n$ are distinct, and
- (3) $(\forall i)(1 \leq i \leq n-1)(\forall x \in A_i - \{a_i\}) [c(\{x, a_i\}) = c_i]$.

Let

$$c_n = \mu d [|\{x : c(\{x, a_n\}) = d\} \cap A_{n-1}| = \infty]$$

(exists since A_{n-1} is infinite),

$$A_n = \{x : c(\{x, a_n\}) = c_n\} \cap A_{n-1}$$

(is infinite by the choice of c_n),

$$a_{n+1} = \mu x [x \in A_n - \{a_1, \dots, a_n\}]$$

(exists since A_n is infinite).

It is easy to see that these values satisfy (1), (2) and (3) above. Let d be the least color that appears infinitely often in the sequence $c_1, c_2, \dots$. Let $A = \{a_i : c_i = d\}$. It is easy to see that A is d -homogeneous. $\square$

The proof of Theorem 3.3 given above is noneffective. To see if the proof could have been made effective, we look at the following potential analogue.

Potential Analogue 3.4 *There is a recursive algorithm $\mathcal{A}$ that performs the following. Given an index e for a recursive 2-coloring of $[N]^2$, $\mathcal{A}$ outputs an index for a recursive homogeneous set. A consequence is that all recursive 2-colorings of $[N]^2$ induce a recursive homogeneous set.*

Specker [161] showed that this Potential Analogue is false. We present a simpler proof by Jockusch [87]. We then show that some recursion-theoretic modifications are true and are the best possible. In particular, we will show the following results, all due to Jockusch.

- (1) There exists a recursive 2-coloring of $[N]^2$ such that no homogeneous set is r.e..
- (2) There exists a recursive 2-coloring c of $[N]^2$ such that no homogeneous set is recursive in K . This coloring c also induces no Σ_2 homogeneous sets.
- (3) Every recursive 2-coloring of $[N]^2$ induces a Π_2 homogeneous set. This is the best possible in terms of the arithmetic hierarchy (by (2)).
- (4) Every recursive 2-coloring of $[N]^2$ induces a homogeneous set A such that $A' \leq_T \emptyset''$.

3.2 Recursive analogue is false

Definition 3.5 A set A is *bi-immune* if neither A nor $\overline{A}$ has an infinite r.e. subset.

Theorem 3.6 *There exists c , a recursive 2-coloring of $[N]^2$, such that no homogeneous set is r.e..*

Proof. Let X be a bi-immune set such that $X \leq_T K$ (such is easily constructed by an initial-segment argument). By the limit lemma (see [159, p. 57]) there exists a 0-1 valued recursive function $f(x, s)$ such that $X(x) = \lim_{s \rightarrow \infty} f(x, s)$. Let c be the following coloring:

$$c(\{a, b\}) = \begin{cases} f(a, b) + 1 & \text{if } a < b; \\ f(b, a) + 1 & \text{if } b < a. \end{cases}$$

(The purpose of the '+1' is to ensure that $\text{range}(c) \subseteq \{1, 2\}$.)

It is easy to see that a 1-homogeneous set is an infinite subset of $\overline{X}$, and a 2-homogeneous set is an infinite subset of X . Since X is bi-immune, there are no infinite r.e. subsets of X or $\overline{X}$, hence there are no r.e. infinite homogeneous sets. $\square$

Theorem 3.7 *There exists c , a recursive 2-coloring of $[\mathbb{N}]^2$, such that no homogeneous set is recursive in K . This coloring also has no Σ_2 homogeneous sets.*

Proof. By the limit lemma, for every set $A \leq_T K$ there exists a 0-1 valued primitive recursive f such that $A(x) = \lim_{s \rightarrow \infty} f(x, s)$. Let $f_0, f_1, f_2, \dots$ be a standard enumeration of all 2-place 0-1 valued primitive recursive functions. Let A_i be the partial function defined by $A_i(x) = \lim_{s \rightarrow \infty} f_i(x, s)$ (if for x the limit does not exist, then $A_i(x)$ is not defined). Note that every set recursive in K is some A_i . We use A_i to represent both the set and its characteristic function.

We construct c , a recursive 2-coloring of $[\mathbb{N}]^2$, to satisfy the following requirements:

$$R_e : A_e \text{ total, infinite} \Rightarrow (\exists x, y, z \in A_e) \\ [x, y, z \text{ distinct and } c(\{x, z\}) \neq c(\{y, z\})].$$

It is easy to see that such a c will have no homogeneous set $A \leq_T K$. Let $A_e^s = \{x : f_e(x, s) = 1\} \cap \{0, 1, \dots, s-1\}$.

Construction

STAGE s :

In this stage we will determine $c(\{s, 0\}), c(\{s, 1\}), \dots, c(\{s, s-1\})$, and try to satisfy $R_0, \dots, R_s$. For each R_e , $0 \leq e \leq s$, in turn we look for the least two elements $x, y \in A_e^s$ such that $c(\{s, x\})$ and $c(\{s, y\})$ are not yet determined. If such an x and y exist, then set $c(\{s, x\}) = 1$ and $c(\{s, y\}) = 2$. After all the requirements are considered, set $c(\{s, x\}) = 1$ for all $x \leq s-1$ such that $c(\{s, x\})$ is not determined. End of the construction.

It is clear that c is recursive. We show that each R_e is satisfied. Assume A_e is total and infinite. Let $x_1 < x_2 < \dots < x_{2e+2}$ be the first $2e+2$ elements of A_e . Let s' be such that, for all $y \leq x_{2e+2}$ and for all $t \geq s'$, $f_e(y, t) = A_e(y)$.

Let s be the least stage such that $s \geq s'$ and $s \in A_e$. At stage s , when considering R_e , at most $2e$ of the pairs $\{s, x_i\}$ will have been colored (each $R_{e'}$, $0 \leq e' \leq e-1$, colors at most two pairs). Hence there exist $x, y \in \{x_1, \dots, x_{2e+2}\} \subseteq A_e$ such that $c(\{s, x\}) \neq c(\{s, y\})$. Since $x, y, s \in A_e$, R_e is satisfied.

We now show that c has no homogeneous Σ_2 sets. Assume, by way of contradiction, that there is a Σ_2 homogeneous set. Every infinite Σ_2 set has an infinite subset that is recursive in K , hence there exists an infinite subset B (of the homogeneous set) such that $B \leq_T K$. Since an infinite subset of a homogeneous set is homogeneous, B is a homogeneous set. Since $B \leq_T K$, this contradicts the nature of c . $\square$

3.3 How hard is it to tell if a homogeneous set is recursive?

In Section 3.2 we saw that it is possible for a recursive 2-coloring of $[N]^2$ to not induce any recursive homogeneous sets. We examine how hard it is to tell if this is the case. We will show that the problem of determining if a coloring induces a recursive homogeneous set is Σ_3 -complete.

Notation 3.8 Throughout this section, let D be the set of indices for total recursive functions whose range is a subset of $\{1, 2\}$. We interpret elements of D as 2-colorings of $[N]^2$.

Lemma 3.9 *There exists a recursive function f such that:*

- (i) $x \in \text{COF} \Rightarrow \{f(x)\}^K$ decides a finite set,
- (ii) $x \notin \text{COF} \Rightarrow \{f(x)\}^K$ decides a bi-immune set.

Proof. Given x , we ‘try’ to construct a bi-immune set $A \leq_T K$ by an initial-segment argument. If $x \in \text{COF}$, then our attempt will fail and A will be finite; however, if $x \notin \text{COF}$, then our attempt will succeed.

We try to construct A to satisfy the following requirements:

$$\begin{aligned} R_{2e} : W_e \text{ infinite} &\Rightarrow (\exists y \in W_e - A), \\ R_{2e+1} : W_e \text{ infinite} &\Rightarrow (\exists y \in W_e - \bar{A}). \end{aligned}$$

At the end of stage s of the construction, we will have

- (i) A_s , an initial segment of A , and
- (ii) i_s , the index of the next requirement that needs to be satisfied.

Note that if $|A_s| = n$, then A has been determined for $0, 1, \dots, n-1$.

Formally the construction should be of an oracle Turing machine $\{f(x)\}^{(\cdot)}$. Informally, we write it as a construction recursive in K , since the only oracle we use is K .

Construction of $\{f(x)\}^K$

STAGE 0:

$A_0 = \lambda$ (the empty string). $i_0 = 0$.

STAGE $s+1$:

Ask K “ $s \in W_x$?”. If YES, then set $A_{s+1} = A_s 0$ and $i_{s+1} = i_s$, and go to the next stage. If NO, then we work on satisfying R_{i_s} . There are two cases, depending on whether i_s is even or odd.

Case $i_s = 2e$. Then do the following: Ask K “ $(\exists y \in W_e)[y \geq |A_s|]$?”, so $A(y)$ has not yet been determined. If NO, then W_e is finite, so R_{2e} is satisfied, hence we set $A_{s+1} = A_s 0$ and $i_{s+1} = i_s + 1$. If YES, then we set $A_{s+1} = A_s 0^{y+1-|A_s|}$, so $A(y) = 0$, and $i_{s+1} = i_s + 1$.

Case $i_s = 2e + 1$. Then do the following. Ask K “ $(\exists y \in W_e)[y \geq |A_s|]$?”, so $A(y)$ has not yet been determined. If NO, then W_e is finite, so R_{2e} is satisfied, hence we set $A_{s+1} = A_s 0$ and $i_{s+1} = i_s + 1$. If YES, then we set $A_{s+1} = A_s 0^{y-|A_s|} 1$, so $A(y) = 1$, and $i_{s+1} = i_s + 1$. End of the construction.

If $x \in \text{COF}$, then for almost all s we merely append 0 to A_s . Hence A is finite, and $\{f(x)\}^K$ decides a finite set.

If $x \notin \text{COF}$, then for every i , requirement R_i is satisfied at stage $s+1$, where s is the $(i+1)$ -st element of $\overline{W}_x$. Hence all requirements are satisfied, A is bi-immune, and $\{f(x)\}^K$ decides a bi-immune set. $\square$

Theorem 3.10 *The set*

$\text{RECRAM} = \{e : e \in \text{TWOCOL} \text{ and } \{e\} \text{ induces a recursive homog. set}\}$

is Σ_3 -complete.

Proof. RECRAM consists of all $e \in \text{TWOCOL}$ such that there exist i, d with the following properties:

- (1) $i \in \text{TOT01}$ and $d \in \{1, 2\}$,
- (2) $(\exists^\infty x)\{i\}(x) = 1$, and
- (3) $(\forall x, y) [(\{i\}(x) = 1 \wedge \{i\}(y) = 1 \wedge x \neq y) \Rightarrow \{e\}(x, y) = d]$.

Clearly RECRAM is Σ_3 . To show that RECRAM is Σ_3 -hard we show that $\text{COF} \leq_m \text{RECRAM}$.

Given x , we produce an index for a recursive coloring of $\mathbb{N}^2$ such that $x \in \text{COF}$ iff that coloring has a recursive homogeneous set. Let f be defined as in Lemma 3.9. Let A_x be the set decided by $\{f(x)\}^K$, and let g be the total recursive 0-1 valued function such that

$$A_x(z) = \lim_{s \rightarrow \infty} g(z, s),$$

(g exists by the limit lemma [159, p. 57].) Let c be the coloring defined by

$$c(\{a, b\}) = \begin{cases} g(a, b) + 1 & \text{if } a < b, \\ g(b, a) + 1 & \text{if } b < a. \end{cases}$$

(The purpose of the '+1' is to ensure that $\text{range}(c) \subseteq \{1, 2\}$.) Let e be the index of this coloring. Note that e can be found effectively from x .

If $x \notin \text{COF}$, then A_x is bi-immune, so c is identical to the coloring in Theorem 3.6. Hence, by the reasoning used there, no homogeneous set can be Σ_1 . Hence no homogeneous set can be recursive.

If $x \in \text{COF}$, then A_x is finite. We show that in this case there is a recursive homogeneous set, by defining a recursive increasing function h whose range is homogeneous. Let a be the least number such that none of the numbers $a, a+1, a+2, \dots$ are in A_x .

$$h(0) = a$$

$$h(n+1) = \mu b \left[\bigwedge_{i=0}^n (b > h(i) \wedge g(h(i), b) = 0) \right]$$

By induction one can show that $h(n)$ is always defined. It is easy to see that the set of elements in the range of h forms a homogeneous set. $\square$

3.4 Recursion-theoretic modifications

Theorem 3.11 *If c is a recursive k -coloring of $[\mathbb{N}]^2$, then there exists a Π_2 homogeneous set.*

Proof. We prove the $k = 2$ case. The general case is similar.

We essentially reprove Ramsey's Theorem carefully so that the homogeneous set is Π_2 (the usual proof yields a homogeneous set that is $\leq_T \emptyset''$).

We construct a sequence of numbers $a_1 < a_2 < \dots < a_i \dots$ and a sequence of colors $c_1, c_2, \dots, c_i, \dots$ (we think of a_i as being colored c_i). The set R of red numbers will be Π_2 in any case, but possibly finite. If R is infinite, then it will be the homogeneous Π_2 set that we seek. If R is finite, then the set B of blue numbers will be Π_2 and infinite; hence it will be our desired Π_2 set.

We approximate the a_i 's and c_i 's in stages by a_i^s and c_i^s . We prove that both $\lim_{s \rightarrow \infty} a_i^s$ and $\lim_{s \rightarrow \infty} c_i^s$ exist.

By coloring a_i^s by c_i^s , for $1 \leq i \leq k$, we are guessing that there is an infinite number of n such that, for all $i \leq k$, $c(\{a_i^s, n\}) = c_i^s$. We will initially guess that a number is colored RED, but we may change our minds.

Construction

STAGE 0:

Set $a_1^0 = 0$ and color it RED, i.e. $c_1^0 = \text{RED}$.

STAGE $s + 1$:

We have $a_1^s < \dots < a_k^s$ (some k) and we want to extend it. Let M be larger than any number that has ever been an a_i^t for any $t \leq s$. Ask the following question (recursive in K): "Does there exist $n > M$ such that, for all $i \leq k$, $c(\{a_i^s, n\}) = c_i^s$?"

If the answer is YES, then look for n until you find it. Set $a_{k+1}^{s+1} = n$ and $c_{k+1}^{s+1} = \text{RED}$; and for all $i \leq k$, set $a_i^{s+1} = a_i^s$ and $c_i^{s+1} = c_i^s$.

If the answer is NO, then by a series of similar questions find the value of $\max \{m : m \leq k - 1 \wedge (\exists n > M)(\forall i \leq m)[c(\{a_i^s, n\}) = c_i^s]\}$ (the value $m = 0$ is permitted). Denote this number by m . Let $n > M$ be the least number such that $(\forall i \leq m)[c(\{a_i^s, n\}) = c_i^s]$. Note that since m was maximum, the statement " $(\forall i \leq m + 1)[c(\{a_i^s, n\}) = c_i^s]$ " is false. Hence $c(\{a_{m+1}^s, n\}) \neq c_{m+1}^s$. We will keep a_{m+1}^s but change its color, and discard all a_i^s with $i \geq m + 2$. Formally,

- (1) set $a_{m+1}^{s+1} = a_{m+1}^s$,
- (2) set c_{m+1}^{s+1} to the opposite of what c_{m+1}^s was,
- (3) for $i \leq m$, set $a_i^{s+1} = a_i^s$ and $c_i^{s+1} = c_i^s$, and
- (4) for $i \geq m+2$, a_i^{s+1} and c_i^{s+1} are undefined.

The numbers a_i^s for $i \geq m+2$ are called *discarded*. End of the construction.

We show that

- (i) the sequences of a_i 's and c_i 's reach limits, and
- (ii) either the set of RED numbers or the set of BLUE numbers is a Π_2 homogeneous set.

Claim 1. For all e , $\lim_{s \rightarrow \infty} a_e^s$ exists and $\lim_{s \rightarrow \infty} c_e^s$ exists.

Proof of Claim 1. Note that when a value of m is found in the NO case of the construction, then the elements discarded are those in places $m+2$ and larger. This observation will help prove the claim.

We proceed by induction on e . By the above observation, for $e = 1$, a_1^s is never discarded. If ever c_1^s turns BLUE, it is because for almost all x , $c(\{a_1^s, x\})$ is BLUE. Hence it will never change color again.

Assume that the claim is true for $i < e$, hence for $1 \leq i \leq e-1$ there exists $a_i = \lim_{s \rightarrow \infty} a_i^s$ and $c_i = \lim_{s \rightarrow \infty} c_i^s$. Let s be the least number such that all the a_i^s and c_i^s ($1 \leq i \leq e-1$) have settled down, i.e., for all $t \geq s$, $a_i^t = a_i$ and $c_i^t = c_i$. By the end of stage $s+1$, the values of a_e^{s+1} and c_e^{s+1} are defined. During stages $t \geq s$, if the NO case of the construction happens, then the resulting value of m will be $\geq e$. Hence a_e^t will never be discarded, so a_e^s has reached a limit, which we call a_e . Assume a_e changes to BLUE at some stage $t \geq s$. During stage t it was noted that, for almost all n , if for all $i < e$, $c(\{a_i, n\}) = c_i$ then $c(\{a_e, n\}) = \text{BLUE}$. Hence a_e will never change color during a later stage. Hence $c_e = \lim_{s \rightarrow \infty} c_e^s$ exists. $\square$

Claim 2. Let d be either RED or BLUE. If $A = \{a_e : c_e = d\}$ is infinite, then it is homogeneous.

Proof of Claim 2. Let a_e, a_i be elements of A with $e < i$. Let s be the least number such that $a_e^s = a_e$ and $c_e^s = d$. Note that a_i must appear in

the sequence at a stage past s , and that for all $t \geq s$, $a_e^s = a_e^t$ and $c_e^s = c_e^t$. By the construction, it must be the case that $c(\{a_e, a_i\}) = d$. Hence A is homogeneous. $\square$

Claim 3. Let $M = \{a_1, a_2, \dots\}$, $R = \{a_i : c_i = \text{RED}\}$, $B = \{a_i : c_i = \text{BLUE}\}$. The set M is infinite. The sets M and R are Π_2 . If R is finite, then B is infinite and Π_2 .

Proof of Claim 3. By Claim 1 and the construction, M is infinite.

$$x \in \overline{M} \quad \text{iff} \quad (\exists s) \left[\left[\text{at stage } s \text{ the YES case occurred with } n > x \right. \right. \\ \left. \left. \text{and } (\forall i)x \neq a_i^s \right] \vee [x \text{ is discarded at stage } s] \right].$$

The matrix of this formula is recursive in K , so the entire formula can be rewritten in Σ_2 form. Hence M is Π_2 .

$$x \in \overline{R} \quad \text{iff} \quad x \notin M \quad \text{or} \quad (\exists s)[x \text{ changes from RED to BLUE at stage } s].$$

(For x satisfying the second clause, x will either stay in M and be BLUE, or leave M . In either case, x is not RED.)

The second set is Σ_2 , since the matrix of the formula is recursive in K . Hence $\overline{R}$ is the union of two Σ_2 sets, so it is Σ_2 ; therefore, R is Π_2 .

If R is finite then $B = M - R$ is infinite. Note that the equation

$$x \in \overline{B} \quad \text{iff} \quad x \notin M \quad \text{or} \quad x \in R$$

yields a Σ_2 definition of $\overline{B}$, and hence B is Π_2 . $\square$ $\square$

Theorem 3.12 If c is a recursive k -coloring of $[\mathbb{N}]^2$, then there exists a homogeneous set A such that $A' \leq_T \emptyset''$.

Proof. We prove the $k = 2$ case. The general case is similar.

We define a Π_1^0 class of functions $\mathcal{F}$ such that

- (a) $\mathcal{F}$ is nonempty,
- (b) there exists $g \leq_T K$ such that $(\forall f \in \mathcal{F})(\forall n)[f(n) \leq g(n)]$, and
- (c) for every $f \in \mathcal{F}$ there exists $A \leq_T f$ such that A is homogeneous.

The desired theorem will follow from Theorem 2.13. We describe $\mathcal{F}$ by describing a recursive tree T (see Proposition 2.14).

For $\sigma = (\langle a_0, c_0 \rangle, \langle a_1, c_1 \rangle, \dots, \langle a_k, c_k \rangle)$, $\sigma \in T$ iff

- (1) $a_0 = 0$,
- (2) for all i , $0 \leq i \leq k$, $c_i \in \{1, 2\}$ (we think of the c_i as being colors), and
- (3) for all j , $1 \leq j \leq k$,

$$a_j = \mu x [x > a_{j-1} \wedge (\forall i \leq j-1) c(\{a_i, x\}) = c_i].$$

It is clear that testing $\sigma \in T$ is recursive. By a noneffective proof (similar to the construction of $a_0, a_1, \dots$ in the proof of Theorem 3.3) one can show that T has an infinite branch, hence $\mathcal{F}$ is nonempty.

We now define a bounding function $g \leq_T K$. We need two auxiliary functions, NEXT and h . Let $\sigma = (\langle a_0, c_0 \rangle, \dots, \langle a_n, c_n \rangle)$.

$$\text{NEXT}(\sigma) = \begin{cases} \mu x [x > a_n \wedge (\forall i \leq n) c(\{a_i, x\}) = c_i] & \text{if it exists and } \sigma \in T \\ 0 & \text{otherwise.} \end{cases}$$

Note that $\text{NEXT} \leq_T K$; and that if $\sigma = (\langle a_0, c_0 \rangle, \dots, \langle a_n, c_n \rangle)$ and $\sigma \in T$, then either

- (i) $\text{NEXT}(\sigma) = x \neq 0$, hence
 - (a) $(\langle a_0, c_0 \rangle, \dots, \langle a_n, c_n \rangle, \langle x, 1 \rangle) \in T$, and
 - (b) $(\langle a_0, c_0 \rangle, \dots, \langle a_n, c_n \rangle, \langle x, 2 \rangle) \in T$, or
- (ii) $\text{NEXT}(\sigma) = 0$, and there are no extensions of σ in T .

Let h be defined by $h(0) = 1$, and

$$h(n+1) = \max_{\substack{a_1, \dots, a_n \in \{1, 2, \dots, h(n)\}, \\ c_0, \dots, c_n \in \{1, 2\}}} \text{NEXT}(\langle a_0, c_0 \rangle, \dots, \langle a_n, c_n \rangle).$$

Note that $h \leq_T K$, and for all $\sigma \in T$ and for all n , $h(n)$ bounds the first component of $\sigma(n)$.

Let $g(n) = \max_{x \leq h(n)} \{\langle x, 1 \rangle, \langle x, 2 \rangle\}$. Clearly $g \leq_T K$, and for every $\sigma \in T$ and $n \in \mathbb{N}$, $\sigma(n) \leq g(n)$.

Let f be a function defined by an infinite branch of T (i.e., $f \in \mathcal{F}$). Let $c \in \{1, 2\}$ be such that $\exists^\infty n f(n) = \langle a_n, c \rangle$. Let $A = \{a : (\exists i) f(i) = \langle a, c \rangle\}$. The set A is homogeneous by the definition of f . Since f is increasing, $A \leq_T f$. $\square$

Theorem 3.12 has been improved by Cholak, Jockusch and Slaman [38]. We state the theorem but not the proof.

Theorem 3.13 *If c is a recursive k -coloring of $[\mathbb{N}]^2$, then there exists a homogeneous set A such that $A'' \leq_T \emptyset''$.*

The proof uses the result by Jockusch and Stephan [91] that there is a low_2 r -cohesive set, as well as a relativization of a new result of the authors that, if $A_1, A_2, \dots, A_n$ are Δ_2^0 -sets and $\cup_{i=1}^n A_i = \mathbb{N}$, then some A_i has an infinite low_2 subset.

3.5 Connections to proof theory

We have only considered Ramsey's Theorem for coloring $[\mathbb{N}]^2$. The full theorem involves coloring $[\mathbb{N}]^m$.

Theorem 3.14 *Let $m \geq 1$. If c is a k -coloring of $[\mathbb{N}]^m$, then there exists a homogeneous set.*

Sketch Proof. The proof is by induction on m and uses the technique of Theorem 3.3. $\square$

The above theorem will henceforth be referred to as GRT (General Ramsey's Theorem). Jockusch [87] has shown the following three theorems. The proofs are similar to the proofs of Theorems 3.12, 3.11, and 3.7.

Theorem 3.15 *Let $m \geq 1$. If c is a recursive k -coloring of $[\mathbb{N}]^m$, then there exists a homogeneous set A such that $A' \leq_T \emptyset^{(m)}$.*

Theorem 3.16 *Let $m \geq 1$. If c is a recursive k -coloring of $[\mathbb{N}]^m$, then there exists a homogeneous set $A \in \Pi_m$.*

Theorem 3.17 *Let $m \geq 2$. There exists a recursive 2-coloring c of $[\mathbb{N}]^m$ such that no homogeneous set is Σ_m .*

Several people observed that this last result has implications for proof theory [168]. We take a short digression into proof theory so that we can state the observation.

Proof theory deals with the strength required by an axiom system in order to prove a given theorem. Before discussing axioms, we need a language strong enough to state the theorem in question.

Definition 3.18 Let $\mathcal{L}_1$ be the language that contains the usual logical symbols, variables that we intend to range over $\mathbb{N}$, constant symbols 0 and 1, and the usual arithmetic symbols $+$, $<$, $\times$. *Peano Arithmetic* (henceforth PA) is a set of axioms that establish the usual rules of addition and multiplication (e.g., associative) and allow a sentence to be proven by induction. (For details of the axioms see any elementary logic texts.)

PA suffices for most finite mathematics, such as finite combinatorics and number theory. Gödel ([67], but see any elementary logic text for a proof in English) showed that there is a sentence φ that can be stated in $\mathcal{L}_1$, which is true but is not provable in PA. The sentence φ is not ‘natural’, in that it was explicitly designed to have this property. Using this result, Gödel showed that the sentence “PA is consistent” (which is expressible in $\mathcal{L}_1$) is not provable in PA. While this sentence is natural to a logician, it may not be natural to a non-logician. Later in this section we will see sentences that are natural (to a combinatorist) and true, but which are not provable in PA.

Note that GRT cannot be stated in $\mathcal{L}_1$. Hence we go to a richer language and axiom system.

Definition 3.19 Let $\mathcal{L}_2$ be $\mathcal{L}_1$, together with a second type of variable (for which we use capital letters) that is intended to range over subsets of $\mathbb{N}$. Let CA (classical analysis) be the axioms of PA together with the following axioms:

- (1) $(\forall A)[(0 \in A \wedge (\forall x)(x \in A \Rightarrow x + 1 \in A)) \Rightarrow (\forall x)(x \in A)]$ (induction).
- (2) For any formula $\varphi(x)$ of $\mathcal{L}_2$ that does not contain the symbol “ A ”, we have the axiom $(\exists A)(\forall x)[x \in A \Leftrightarrow \varphi(x)]$ (comprehension).
- (3) $(\forall A)(\forall B)[(\forall x)(x \in A \Leftrightarrow x \in B) \Rightarrow A = B]$ (extensionality).

The strength of CA can be moderated by restricting the comprehension axiom to certain types of formulas.

Definition 3.20 PPA is CA with the comprehension axiom weakened to use only arithmetic formulas (i.e., formulas with no bound set variables) for φ .

Notation 3.21 If S is an axiom system (e.g., PA) and φ is a sentence, then $S \vdash \varphi$ means that there is a proof of φ from S , and $\neg(S \vdash \varphi)$ means that there is no proof of φ from S .

We can now state and prove a theorem about how hard it is to prove GRT.

Theorem 3.22 $\neg(\text{PPA} \vdash \text{GRT})$.

Sketch Proof. Let $A(m, X, Y)$ be a formula in $\mathcal{L}_2$ with no bound set variables. If

$$\text{PPA} \vdash (\forall m)(\forall X)(\exists Y) A(m, X, Y),$$

then there exists an i such that

$$\text{PPA} \vdash (\forall m)(\forall X)(\exists Y \in \Sigma_i^X) A(m, X, Y).$$

(See [168] for proofs of a similar theorem by Jockusch and Solovay.)

We hope to make

$$(\forall m)(\forall X)(\exists Y) A(m, X, Y)$$

the statement GRT. Let the notation $X \subseteq [\mathbf{N}]^m$ mean that X is a subset of $\mathbf{N}$ that we are interpreting via some fixed recursive bijection of $\mathbf{N}$ to $[\mathbf{N}]^m$ to be a subset of $[\mathbf{N}]^m$. This type of bijection can easily be described in PPA (actually in PA). Note that one can interpret $X \subseteq [\mathbf{N}]^m$ to mean that X represents a 2-coloring of $[\mathbf{N}]^m$ in which all elements of X are RED and all elements not in X are BLUE.

Let $A(m, X, Y)$ be the statement

$$X \subseteq [\mathbf{N}]^m \Rightarrow Y \text{ is homogeneous for the coloring } X.$$

Note that $(\forall m)(\forall X)(\exists Y) A(m, X, Y)$ is GRT. Assume, by way of contradiction, that $\text{PPA} \vdash \text{GRT}$. Then there exists i such that

$$\text{PPA} \vdash (\forall m)(\forall X)(\exists Y \in \Sigma_i^X) A(m, X, Y),$$

hence $(\forall m)(\forall X)(\exists Y \in \Sigma_i^X) A(m, X, Y)$ is true. If X is recursive then $\Sigma_i^X = \Sigma_i$. Hence we obtain

$$(\forall m)(\forall X, X \text{ recursive})(\exists Y \in \Sigma_i) A(m, X, Y). \quad (*)$$

Let $m > i$. By Theorem 3.17 there exists a recursive 2-coloring of $[N]^m$ that has no Σ_m homogeneous sets. This coloring will have no Σ_i homogeneous sets.

$$(\exists m)(\exists X \text{ recursive})(\forall Y \in \Sigma_i) [\neg A(m, X, Y)].$$

This contradicts $(*)$. Hence $\neg(\text{PPA} \vdash \text{GRT})$. $\square$

Recall that Gödel showed there is a true sentence φ that is independent of a natural system (PA), but is unnatural (to a non-logician). By contrast, Theorem 3.22 exhibits a sentence (GRT) that is natural to some mathematicians (combinatorists), but is independent of an unnatural system (PPA). The question arises as to whether there are natural true sentences that are independent of natural systems. The answer is yes!

Paris and Harrington [134] proved the following variant of the finite Ramsey theorem to be independent of PA. Ketonen and Solovay [93] gave a different proof. A scaled down version of the proof is in Graham, Roth and Spencer's book on Ramsey theory [68].

Theorem 3.23 *Let $p, k, m \in \mathbb{N}$. There exists an $n = n(p, k, m)$ such that for all k -colorings of $[1, 2, \dots, n]^m$, there exists a homogeneous set A such that $|A| \geq p$, and $|A|$ is larger than the least element of A . (Homogeneous is the usual definition, except that the set is not infinite.)*

The independence proof of Paris and Harrington used model theory, while the proof of Ketonen and Solovay showed that the function $n(p, k, m)$ grows too fast to be proven to exist in PA. Both proofs are somewhat difficult. For an easier independence proof of a natural Ramsey-type theorem see Kanamori and McAloon [92]. For more on undecidability of Ramsey-type theorems see [21]. For other examples of natural theorems that are independent of natural systems see [155].

While it might be nice to say, "The independence results obtained by Jockusch were to foreshadow the later ones of Paris-Harrington and others", such a statement would be false. The observation that Jockusch's work on Ramsey theory leads to results on independence of true sentences from PPA happened after, and was inspired by, the Paris-Harrington results.

3.6 Miscellaneous

3.6.1 2-colorings of $[N]^\omega$

There are versions of Ramsey's theorem that involve well-behaved colorings of $[N]^\omega$, instead of arbitrary colorings of $[N]^k$. Note that $[N]^\omega$ is the set of all infinite subsets of N . View every element of $[N]^\omega$ as an element of $\{0, 1\}^\omega$, the characteristic string of the set represented by that element. View $[N]^\omega$ as a topological space, where X is a basic open set iff there exists $\sigma \in \{0, 1\}^*$ such that $X = \{f \in [N]^\omega \mid \sigma \prec f\}$. A coloring $c : [N]^\omega \rightarrow \{1, \dots, m\}$ is *clopen* (*Borel*, *analytic*) if there is an i such that $c^{-1}(i)$ is clopen (Borel, analytic). Recall that clopen (in topology) means that both a set and its complement are the union of basic open sets.

We call a 2-coloring of $[N]^\omega$ *Ramsey* if it induces a homogeneous set. Galvin and Prikry [60] showed that all Borel colorings are Ramsey. Silver [150] and Mathias [122] improved this by showing that all Σ_1^1 colorings are Ramsey. This result is essentially optimal, since there are models of set theory which have $\Sigma_2^1 \cap \Pi_2^1$ colorings that are not Ramsey (this happens when $V = L$); and there are models of set theory where all Π_2^1 colorings are Ramsey (this happens when there is a measurable cardinal [150]). An easier proof of the Silver-Mathias result was given by Ellentuck [50]. A game-theoretic proof that uses determinacy was given by Tanaka [163]. See [26] for a summary of this area.

A special case of the Galvin-Prikry theorem is that clopen colorings are Ramsey. A 2-coloring $c : [N]^\omega \rightarrow \{1, 2\}$ is *recursively clopen* if both $c^{-1}(1)$ and $c^{-1}(2)$ can be described as the set of extensions of some recursive subset of $[N]^{<\omega}$. Simpson [151] showed that the recursive version of the clopen-Galvin-Prikry theorem is false in a strong way. He showed that for every recursive ordinal α there exists a recursively clopen 2-coloring such that, if A is a homogeneous set, then $\emptyset^\alpha \leq_T A$. Solovay [160] showed that every recursively clopen 2-coloring induces a hyperarithmetic homogeneous set. Clote [41] refined these results by looking at the order type of colorings.

3.6.2 Almost homogeneous sets

If c is a 2-coloring of $[N]^2$, then an infinite set $A \subseteq N$ is *almost homogeneous* if there exists a finite set F such that $A - F$ is homogeneous. A 2-coloring c of $[N]^2$ is r.e. if either $c^{-1}(\text{RED})$ or $c^{-1}(\text{BLUE})$ is r.e.. An infinite set A

is *m-cohesive* if, for every r.e. coloring of $[N]^m$, A is almost homogeneous (1-cohesive is the same as cohesive).

From the definition it is not obvious that there are sets that are *m-cohesive*. We show, in a purely combinatorial way (no recursion theory), that 2-cohesive sets exist (*m-cohesive* is similar). Let $c_1, c_2, \dots$ be the list of all r.e. 2-colorings (for this proof they could be any countable class of 2-colorings that is closed under finite variations). Let $A_0 = N$ and, for all $i > 0$, let A_i be a subset of A_{i-1} which is homogeneous with respect to c_i . Let a_i be the least element of $A_i - \{a_0, \dots, a_{i-1}\}$. The set $\{a_0, a_1, \dots\}$ is 2-cohesive.

From the above proof it is not obvious that r.e. colorings have arithmetic 2-cohesive sets. Slaman [157] showed that there are Π_3 2-cohesive sets. Jockusch [88] showed that there are Π_2 2-cohesive sets. His proof combined the Friedberg-Yates technique used to construct a maximal set (see [159, Theorem X.3.3]) with Jockusch's technique used to prove Theorem 3.11 (of this paper). This result is optimal in the sense that there is no Σ_2 2-cohesive set (by Theorem 3.7).

Hummel [84] is investigating, in her thesis, properties of *k-cohesive* sets. She has shown that there exists an r.e. 2-coloring c of $[N]^2$ (i.e., one of $c^{-1}(\text{RED})$ or $c^{-1}(\text{BLUE})$ is r.e.) such that if B is a homogeneous set then $\emptyset' \leq_T B$; hence, if A is a 2-cohesive set, then $\emptyset' \leq_T A$. She has also shown that there exists a recursive 2-coloring of $[N]^2$ such that if B is a Π_2^0 homogeneous set, then $\emptyset'' \leq_T B \oplus \emptyset'$; hence, if A is a 2-cohesive Π_2^0 set, then $A \equiv_T \emptyset''$. The construction is the same as that in Theorem 3.7 used to obtain a 2-coloring such that, if A is homogeneous, then $A \not\leq_T K$. The proof that this construction yields this stronger result uses a result of Hummel which is analogous to Martin's result [121] that effectively simple sets are complete.

3.6.3 Degrees of homogeneous sets

By Theorem 3.7 there are 2-colorings of $[N]^2$ that induce no recursive-in- K homogeneous sets. The question arises as to whether there are 2-colorings of $[N]^2$ such that, for all homogeneous sets A , $K \leq_T A$. There is such a 2-coloring of $[N]^3$: color $\{a < b < c\}$ RED if $(\exists x < a)[x \in K_c - K_b]$, and BLUE otherwise, where K_b (K_c) is the set containing the first b (c) elements of K in some fixed recursive enumeration. This result first appeared in [87] with a different proof. The proof here was communicated to me by Jockusch.

David Seetapun (reported in [84]) has shown that there is no such coloring of $[N]^2$. Formally, he has shown that for any recursive 2-coloring of $[N]^2$ there is a homogeneous set in which K is not recursive. The proof uses a forcing argument.

This problem arose because of possible implications in proof theory. In Paris's original paper on theorems unprovable in PA (see [133]), he has a version of Ramsey's theorem for 2-colorings of $[N]^3$, that is not provable in PA. It uses ideas like the ones in the proof above about 2-coloring $[N]^3$. If the above problem had been solved in the affirmative then there might be a version of Ramsey's theorem about 2-colorings of $[N]^2$, that is not provable in PA. By contrast, Seetapun's result shows that Ramsey's theorem for $[N]^2$ is not equivalent to arithmetic comprehension over RCA_0 . (RCA_0 is a weak fragment of second order arithmetic where (roughly) only recursive sets can be proven to exist. RCA_0 stands for "Recursive Comprehension Axiom". See [152] for an exposition.)

Seetapun and Slaman [148] have shown that given any sequence of non-recursive sets $C_0, C_1, \dots$ and any k -coloring of $[N]^2$, there is an infinite homogenous set H , such that $(\forall i)[C_i \not\subseteq_T H]$. This result has implications for the proof-theoretic strength of Ramsey's Theorem.

3.6.4 Dual Ramsey Theorem

There is a dual version of Ramsey's theorem that involves coloring the 2-colorings themselves. The Dual Ramsey Theorem is proven by Carlson and Simpson [25], and examined recursion-theoretically by Simpson [154].

3.6.5 Ramsey Theory and Peano Arithmetic

Clote [42] has investigated a version of Ramsey's theorem for coloring initial segments I of a model of Peano Arithmetic. He has constructed m -colorings of $[I]^n$ that have no Σ_n weakly-homogeneous subsets (this term is defined in his paper). From this he derives independence results for Peano Arithmetic.

4 Coloring infinite graphs

We consider vertex colorings of infinite graphs. We present the theorem that a graph is k -colorable iff all of its finite subgraphs are k -colorable (originally due to de Bruijn and Erdos [23]), show that a recursive analogue is false, then

show a true combinatorial modification showing that it cannot be improved. We then show that there is true recursion-theoretic modification, and finally state some miscellaneous results.

4.1 Definitions and classical version

Definition 4.1 A graph $G = (V, E)$ is a set V (called *vertices*) together with a set E of unordered pairs of vertices (called *edges*). Edges of the form $\{v, v\}$ are not allowed. If $v \in V$ then the *degree of v in G* is $|\{x \in V : \{v, x\} \in E\}|$.

Definition 4.2 Let $G = (V, E)$ be a graph and $k \geq 1$. A k -coloring of G is a function c from V to $\{1, 2, \dots, k\}$ such that no two adjacent vertices are assigned the same value. The values $\{1, 2, \dots, k\}$ are commonly called *colors*. The *chromatic number of G* , denoted $\chi(G)$, is the minimal k such that G is k -colorable. If no such k exists, then by convention $\chi(G) = \infty$. If $G = (\emptyset, \emptyset)$, then by convention $\chi(G) = 0$. This is the only graph that is 0-colorable. A graph is *colorable* if there exists a $k \in \mathbb{N}$ such that $\chi(G) = k$.

We show that G is k -colorable iff every finite subgraph of G is k -colorable. We give a direct proof of the theorem; it can also be proven by König's Lemma (Theorem 2.3).

Theorem 4.3 Let $G = (V, E)$ be a countable graph. G is k -colorable iff every finite subgraph of G is k -colorable.

Proof. Assume every finite subgraph of G is k -colorable. Assume $V = \mathbb{N}$. Let $G_i = (V_i, E_i)$ be the finite graph defined by $V_i = \{0, 1, \dots, i\}$ and $E_i = [\{0, 1, \dots, i\}]^2 \cap E$. Since G_i is a finite subgraph of G , G_i is k -colorable. Let c_i be a k -coloring of G_i that uses $\{1, 2, \dots, k\}$ for its colors. We use the c_i to define a k -coloring c of G . Let

$$\begin{aligned} c(0) &= \mu x [(\exists^\infty i) c_i(0) = x] \\ c(n+1) &= \mu x [(\exists^\infty i) ((\bigwedge_{y=0}^n c_i(y) = c(y)) \wedge (c_i(n+1) = x))] \end{aligned}$$

It is easy to see that c is a k -coloring of G . The converse is obvious. $\square$

The proof of Theorem 4.3 given above is noneffective. To see if the proof could have been made effective we will look at a potential analogue. In order to state this analogue we need some definitions.

Definition 4.4 A graph $G = (V, E)$ is *recursive* if $V \subseteq \mathbb{N}$ and $E \subseteq [\mathbb{N}]^2$ are recursive.

Definition 4.5 A graph $G = (V, E)$ is *highly recursive* if every vertex of G has finite degree, and the function that maps a vertex to an encoding of the set of its neighbors is recursive.

Definition 4.6 Let $G = (V, E)$ be a graph such that $V \subseteq \mathbb{N}$, and let $k \geq 1$ (in practice G will be a recursive or highly recursive graph). G is *recursively k -colorable* if there exists a recursive function c that is a k -coloring of G . The *recursive chromatic number of G* , denoted by $\chi^r(G)$, is the minimal k such that G is recursively k -colorable. If no such k exists, then by convention $\chi^r(G) = \infty$. If $G = (\emptyset, \emptyset)$, then by convention $\chi^r(G) = 0$. This is the only graph that is recursively 0-colorable. A graph is *recursively colorable* if there exists a $k \in \mathbb{N}$ such that $\chi^r(G) = k$.

We will represent recursive graphs by the Turing machines that determine their vertex and edge sets. An index for a recursive graph will be an ordered pair, the first component of which is an index for a Turing machine which decides the vertex set, the second the edge set.

Definition 4.7 A number $e = \langle e_1, e_2 \rangle$ determines a recursive graph if $e_1, e_2 \in \text{TOT01}$. The graph that $\langle e_1, e_2 \rangle$ determines, denoted by G_e^r , has vertex set

$$V = \{x : \{e_1\}(x) = 1\}$$

and edge set

$$E = \{\{x, y\} : x, y \in V, x \neq y, \{e_2\}(x, y) = \{e_2\}(y, x) = 1\}.$$

Definition 4.8 A number $\langle e_1, e_2 \rangle$ determines a highly recursive graph if $e_1 \in \text{TOT01}$, $e_2 \in \text{TOT}$; and, if $\{e_2\}$ is interpreted as mapping numbers to finite sets of numbers, then $x \in \{e_2\}(y)$ iff $y \in \{e_2\}(x)$. The graph that $\langle e_1, e_2 \rangle$ determines, denoted by G_e^{hr} , has vertex set

$$V = \{x : \{e_1\}(x) = 1\}$$

and edge set

$$E = \{\{x, y\} : x, y \in V, x \neq y, x \in \{e_2\}(y)\}.$$

Potential Analogue 4.9 There is a recursive algorithm $\mathcal{A}$ that performs the following. Given

- (1) an index e for recursive graph G_e^r , and
- (2) an index i for a recursive function that will k -color any finite subgraph of G_e^r ,

$\mathcal{A}$ outputs an index for a recursive k -coloring of G_e^r . A consequence is that every recursive k -colorable graph would be recursively k -colorable. (A similar analogue and consequence can be stated for highly recursive graphs.)

We will soon (Theorem 4.15) see that this potential analogue is false. Hence, we will look at modifications of it. There are two parameters to relax: either we can settle for a recursive $f(k)$ -coloring (f is some function) instead of a k -coloring, or we can settle for a k -coloring that is not that strong in terms of Turing degree (the coloring will turn out to be of low Turing degree).

We will show the following.

- (1) There exists a recursive graph G such that $\chi(G) = 2$, but $\chi^r(G) = \infty$. Hence Potential Analogue 4.9 is false. Moreover, no modification allowing more colors is true.
- (2) There exists a highly recursive graph G such that $\chi(G) = k$, but $\chi^r(G) = 2k - 1$. Hence Potential Analogue 4.9 is false even for highly recursive graphs.
- (3) If G is a highly recursive graph that is k -colorable, then one can effectively produce (given an index for G) a recursive $(2k - 1)$ -coloring of G . Hence a combinatorial modification of Potential Analogue 4.9 is true.
- (4) If G is a recursive graph that is k -colorable, there is a k -coloring of low degree. Hence a recursion-theoretic modification of Potential Analogue 4.9 is true.

We will need the following definitions from graph theory.

Definition 4.10 Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be graphs such that $V_1, V_2 \subseteq \mathbf{N}$ (in practice they will be recursive graphs). G_1 and G_2 are *isomorphic* if there is a map from V_1 to V_2 that preserves edges. We denote this by $G_1 \cong G_2$.

Notation 4.11 Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be graphs. $G_1 \subseteq G_2$ means that $V_1 \subseteq V_2$ and $E_1 \subseteq E_2$. $G = G_1 \cup G_2$ is the graph with vertex set $V_1 \cup V_2$ and edge set $E_1 \cup E_2$.

4.2 Recursive analogue is false for recursive graphs

We show that for all a, b such that $2 \leq a < b \leq \infty$, there exists a recursive graph G such that $\chi(G) = a$ and $\chi^r(G) = b$. The lemmas we prove are more general than we need for this application, however they will be used again in Section 4.3.

Definition 4.12 If $\{e\}$ is a Turing machine and W is a set such that, for all $x \in W$, $\{e\}(x) \downarrow$, then $\{e\}(W) = \{\{e\}(x) : x \in W\}$.

The next lemma is implicit in [11], though it was stated and proven in [14].

Lemma 4.13 *Let $i \geq 0$, $\{e\}$ be a Turing machine, and X be an infinite recursive set. There exists a finite sequence of finite graphs $L_1, \dots, L_r$ such that the following conditions hold. For notation, $L_j = (V_j, E_j)$.*

- (i) L_1 is a graph consisting of 2^i isolated vertices. For every j , $2 \leq j \leq r$, $V_{j-1} \subseteq V_j$ and $E_{j-1} \subseteq E_j$. For each j , $1 \leq j \leq r$,
 - (1) $V_j \subseteq X$, and
 - (2) canonical indices for the finite sets V_j and E_j can be effectively computed given e, i, j and an index for X .
- (ii) For every j , $1 \leq j < r$,
 - (1) for all $x \in V_j$, $\{e\}(x) \downarrow$, and
 - (2) L_{j+1} can be obtained recursively from L_j and the values of $\{e\}(x)$ for every $x \in V_j$.
- (iii) There exists a nonempty set $W \subseteq V_r$ of vertices such that either
 - (1) $\{e\}$ is not total on W , so $\{e\}$ is not a coloring of L_r , or
 - (2) there exist $v \in V_r$ and $w \in W$ such that $\{v, w\} \in E_r$ and $\{e\}(v) = \{e\}(w)$, so $\{e\}$ is not a coloring of L_r , or
 - (3) for all $x \in W$, $\{e\}(x) \downarrow$ and $|\{e\}(W)| = i + 1$, so $\{e\}$ is not an i -coloring of L_r .

- (iv) *There is a 2-coloring of L_r in which W is 1-colored. Hence $\chi(L_r) \leq 2$.*
- (v) *An index for a recursive $(i+1)$ -coloring of L_j can be effectively obtained from e, i, j and an index for X .*
- (vi) *L_r is planar.*

The set W witnesses the fact that $\{e\}$ is not an i -coloring of L_r . We call W a witness of type 1, 2, or 3, depending on which subcase of (iii) it falls under. If it falls under more than one, then we take the least such subcase.

Proof. The Turing machine $\{e\}$ is fixed throughout this proof.

We prove this lemma by induction on i . Assume $i = 0$ and x is the first element of X . Let $L_1 = L_r = (\{x\}, \emptyset)$ and $W = \{x\}$. If $\{e\}(x) \uparrow$, then W is a witness of type 1. If $\{e\}(x) \downarrow$, then W is a witness of type 3. In either case conditions (i)–(vi) are easily seen to be satisfied.

Assume this lemma is true for i . We show it is true for $i + 1$. Let $X = Y \cup Z$ be a recursive partition of X into infinite recursive sets such that indices for Y and Z can be obtained from indices for X . Apply the induction hypothesis to the values i, e, Y and also to i, e, Z to obtain the following:

- (a) a sequence of graphs $L_{11}, L_{21}, \dots, L_{r_11}$, and a set W_1 , such that the sequence together with witness set W_1 satisfies (i)–(vi) (note that all the vertices are in Y), and
- (b) a sequence of graphs $L_{12}, L_{22}, \dots, L_{r_22}$, and a set W_2 , such that the sequence together with witness set W_2 satisfies (i)–(vi) (note that all the vertices are in Z).

Assume $r_1 \leq r_2$. We define graphs $L_1, L_2, \dots, L_{r'}$ that satisfy the theorem (r' will be either r_1, r_2 or $r_2 + 1$). For $1 \leq j \leq r_1$ let

$$L_j = L_{j1} \cup L_{j2}.$$

If for all x , x a vertex of L_{r_11} and $\{e\}(x) \downarrow$, then for $r_1 + 1 \leq j \leq r_2$ let

$$L_j = L_{r_11} \cup L_{j2}.$$

(If this does not occur, then L_{r_1} is the final graph, and W_1 is the witness set.) In this case we obtain witness sets as follows. If W_1 (W_2) is a witness of type 1 or 2, then L_{r_2} is our final graph and $W = W_1$ (W_2). The 2-coloring of the

final graph with the witnesses 1-colored can be obtained by combining such colorings from L_{r_11} and L_{r_22} . It is easy to see that the sequence of graphs and the witness set W all satisfy requirements (i)–(vi).

If both W_1 and W_2 are witnesses of type 3, then there are two cases:

Case 1.

If $\{e\}(W_1) \neq \{e\}(W_2)$, then either there is some element $w \in W_1$ such that $\{e\}(w) \notin \{e\}(W_2)$, or there is some element $w \in W_2$ such that $\{e\}(w) \notin \{e\}(W_1)$. We examine the latter case, the former is similar. Our final graph is L_{r_2} and we let $W = W_1 \cup \{w\}$. By the induction hypothesis and the fact that W_1 is of type 3, $|\{e\}(W_1)| = i + 1$. Since $w \notin W_1$ and $\{e\}(w) \notin \{e\}(W_1)$, $|\{e\}(W_1 \cup \{w\})| = i + 2$. Hence W is a witness of type 3. The 2-coloring of the final graph with the witnesses 1-colored can be obtained by combining such colorings from L_{r_11} and L_{r_22} .

Case 2.

If $\{e\}(W_1) = \{e\}(W_2)$, then let w be the least element of X that is bigger than both any element used so far and the number of steps spent on this construction so far (this is done to make the graph recursive). Let

$$L_{r_2+1} = L_{r_2} \cup \{\{u, w\} : u \in W_1\},$$

$$W = W_2 \cup \{w\}.$$

If $\{e\}(w) \uparrow$, then W is a witness of type 1. If $\{e\}(w) \downarrow \in \{e\}(W_1)$, then since w is connected to all vertices in W_1 , W is a witness of type 2. If $\{e\}(w) \downarrow \notin \{e\}(W_1)$, and hence $\{e\}(w) \notin \{e\}(W_2)$, then

$$\{e\}(W) = \{e\}(W_2 \cup \{w\}) = \{e\}(W_2) \cup \{e\}(w),$$

which has cardinality $i + 2$; hence W is a witness of type 3. Hence W is a witness set. A 2-coloring of L_{r_2+1} with W 1-colored can easily be obtained from the 2-coloring of L_{r_11} (that 1-colors W_1) and the 2-coloring of L_{r_22} (that 1-colors W_2).

It is easy to see that the sequence $L_1, L_2, \dots, L_r$, and the set W satisfy (i)–(iv). Given e, i, j and an index for X , one can effectively find indices for Y, Z , and then use the induction hypothesis and the construction to obtain an index for an $(i + 1)$ -coloring of L_j ; hence v holds. $\square$

Lemma 4.14 *Let $a \geq 2$, $i \geq a$, $\{e\}$ be a Turing machine, and X be an infinite recursive set. There exists a finite sequence of finite graphs $L_1, \dots, L_r$ such that the following conditions hold. For notation, $L_j = (V_j, E_j)$.*

(i) *Every L_j is the union of an a -clique $(A, [A]^2)$ and a 2-colorable graph. For every j with $2 \leq j \leq r$, $V_{j-1} \subseteq V_j$ and $E_{j-1} \subseteq E_j$. For every j with $1 \leq j \leq r$,*

(1) $V_j \subseteq X$, and

(2) *canonical indices for the finite sets V_j and E_j can be effectively computed given e, i, j and an index for X .*

(ii) *For every j with $1 \leq j < r$, L_{j+1} can be obtained recursively from L_j and the values of $\{e\}(x)$ for every $x \in V_j - A$.*

(iii) $\chi(L_r) = a$ (this is the difference between this lemma and Lemma 4.13).

(iv) *An index for a recursive $(i+1)$ -coloring of L_j can be effectively obtained from a, e, i, j and an index for X .*

(v) *Every L_j is the union of an a -clique and a planar graph.*

Proof. Let $x_1, \dots, x_a$ be the first a elements of X . Let K_a be the a -clique on the vertices $\{x_1, \dots, x_a\}$. Let $L'_1, \dots, L'_r$ be the sequence obtained from applying Lemma 4.13 to e, i and $X - \{x_1, \dots, x_a\}$. For all j , $1 \leq j \leq r$, let $L_j = L'_j \cup K_a$. $\square$

Theorem 4.15 *Let a, b be such that $2 \leq a < b \leq \infty$. Let X be an infinite recursive set. There exists a recursive graph $G = (V, E)$ such that $\chi(G) = a$, $\chi^r(G) = b$, and $V \subseteq X$. If $a \leq 4$ then G can be taken to be planar.*

Proof. Recursively partition X into sets $U_{\langle e, i \rangle}$ such that $U_{\langle e, i \rangle}$ is infinite. Let $G(e, i)$ be the graph constructed in Lemma 4.14 using parameters a, e, i and $U_{\langle e, i \rangle}$, i.e., $G(e, i)$ is the graph called " L_r ". Let $G = \bigcup_{e=0}^{\infty} \bigcup_{0 \leq i < b} G(e, i)$. Clearly G is recursive and $\chi(G) = a$. Since

$$(\forall e)(\forall i < b) [\{e\} \text{ is not an } i\text{-coloring of } G(e, i)],$$

we have $\chi^r(G) \geq b$. Since $(\forall e)(\forall i < b) [\chi^r(G(e, i)) \leq i + 1]$ in a uniform way, $\chi^r(G) \leq b$. Combining these inequalities yields $\chi^r(G) = b$. $\square$

Corollary 4.16 *There exists a recursive graph G such that $\chi(G) = 2$, $\chi^r(G) = \infty$, and G is planar.*

Note 4.17 The corollary (and its proof) are essentially due to Bean, who actually showed that there exists a *connected* recursive graph G such that $\chi(G) = 3$ and $\chi^r(G) = \infty$. Both Bean's result and ours are optimal since

- (i) for connected recursive graphs G , $\chi(G) = 2 \Rightarrow \chi^r(G) = 2$, and
- (ii) for recursive graphs G , $\chi(G) \leq 1 \Rightarrow \chi^r(G) = \chi(G)$.

In addition, both the graph in Bean's proof and the graph in Corollary 4.16 are planar.

4.3 How hard is it to determine $\chi^r(G)$?

Theorem 4.15 says that there are recursive graphs G such that $\chi(G)$ and $\chi^r(G)$ are very different. Given a graph, how hard is it to tell if it is of this type? In this section we show that, even if $\chi(G)$ is known and $\chi^r(G)$ is narrowed down to two values, it is Σ_3 -complete to determine $\chi^r(G)$. By contrast, the following promise problem is Π_1 -complete: (D, A) , where

$$D = \{e \mid e \text{ is the index of a recursive graph}\}$$

and

$$A = \{e \in D \mid \text{the graph represented by } e \text{ is } k\text{-colorable}\}.$$

Lemma 4.18 *Let $a \geq 2$, $i \geq a$, $\{e\}$ a Turing machine, and X be an infinite recursive set. There exists an infinite sequence of (not necessarily distinct) finite graphs $H_1, H_2, \dots$ such that the following hold. For notation, $H_s = (V_s, E_s)$.*

- (i) $V_1 \subseteq V_2 \subseteq \dots$.
- (ii) For all s , $V_s \subseteq X$ and $\chi(H_s) = a$.
- (iii) Given a, e, i, s , and an index for X , one can effectively find canonical indices for the finite sets V_s and E_s .
- (iv) There exists a finite graph H and a number t such that

$$(\forall s \geq t) [H_s = H].$$

We call this graph $\lim_{s \rightarrow \infty} H_s$.

- (v) H is not i -colored by $\{e\}$.
- (vi) $\chi(H) = a$.
- (vii) Given a, e, i, s , and an index for X , one can effectively find an index for a recursive $(i+1)$ -coloring of H_s .

Proof. Apply Lemma 4.14 to the parameters a, e, i and X . View the construction of L_r as proceeding in stages where, at each stage, only one more step in the construction is executed. Let H_s be the graph produced at the end of stage s . It is easy to see that (i)–(vii) are satisfied. $\square$

Lemma 4.19 *Let $a \geq 2$, $i \geq a$, $\{e\}$ be a Turing machine, and X be an infinite recursive set. Let $y \in \mathbb{N}$. There exists a recursive graph $G = (V, E)$, which depends on y , such that the following hold.*

- (i) $V \subseteq X$.
- (ii) given a, e, i, y , and an index for X , one can effectively find an index for G .
- (iii) Every component of G is finite.
- (iv) $\chi(G) = a$.
- (v) If $y \notin \text{TOT}$, then
 - (a) G consists of a finite number of finite components, and
 - (b) G is not i -colored by $\{e\}$.
- (vi) If $y \in \text{TOT}$, then
 - (a) G consists of an infinite number of finite components,
 - (b) given a, e, i, y , and an index for X , and $v \in V$, one can effectively find the finite component containing v , and
 - (c) given a, e, i, y , and an index for X , one can find an index for a recursive a -coloring of G (this follows from $\chi(G) = a$ and items (vi)(a) and (b)).

Proof. We consider a, e, i, y , and X , fixed throughout this proof. Let

$$X = \bigcup_{j=0}^{\infty} X_j$$

be a recursive partition of X into an infinite number of infinite recursive sets. Let $H_1(j), H_2(j), \dots$ be the sequence of graphs obtained by applying Lemma 4.18 to parameters a, e, i and X_j . We use these graphs to construct G in stages.

Construction

STAGE 0:

$$G_0 = (\emptyset, \emptyset).$$

STAGE $s + 1$:

Let j_s be the least element that is not in $W_{y,s}$. Let G_{s+1} be $G_s \cup H_s(j_s)$. Note that if $j_s \neq j_{s-1}$ then a new component is started. End of the construction.

Let $G = \bigcup_s G_s$. It is clear that G satisfies (i) and (ii). Since for every j , both $H_s(j)$ and $H = \lim_{s \rightarrow \infty} H_s(j)$ are finite, (iii) holds. By Lemma 4.18, each component of G is a -colorable, therefore G is a -colorable. Hence (iv) holds.

Assume $y \notin \text{TOT}$. Let j be the least element of $\overline{W}_y$. Let t be the least stage such that $0, 1, \dots, j-1 \in W_{y,t}$. For all $s \geq t$, $j_s = j$; therefore G consists of a finite number of graphs of the form $H_{s'}(j')$ (where $s' < t$ and $j' < j$) along with $H = \lim_{s \rightarrow \infty} H_s(j)$. Hence (v) (a) holds. By Lemma 4.18, H is not i -colored by $\{e\}$, hence (v) (b) holds.

Assume $y \in \text{TOT}$. Since $W_y = \mathbb{N}$, $\lim_{s \rightarrow \infty} j_s = \infty$. During every stage s such that $j_s \neq j_{s+1}$, a new component is created; therefore G consists of an infinite number of components. Hence (vi) (a) holds.

To establish (vi) (b) we show, given $v \in V$, how to find all the vertices and edges in the finite component containing v . Run the construction until $j, s \in \mathbb{N}$ are found such that v is a vertex of $H_s(j)$ (this will happen since $v \in V$). Run the construction further until t is found such that $j < j_t$ (this will happen since $y \in \text{TOT}$). The finite component of $H_t(j)$ that contains v is the finite component of G that contains v . $\square$

Theorem 4.20 Let $a, b \in \{2, 3, \dots\} \cup \{\infty\}$ where $a < b$. Let D be the set of indices of recursive graphs with chromatic number a and recursive chromatic number either a or b . Let $\text{RECCOL}_{a,b}$ be the 0-1 valued partial function defined by

$$\text{RECCOL}_{a,b}(e) = \begin{cases} 1 & \text{if } e \in D \text{ and } \chi^r(G_e^r) = a, \\ 0 & \text{if } e \in D \text{ and } \chi^r(G_e^r) = b, \\ \text{undefined} & \text{if } e \notin D. \end{cases}$$

The promise problem $(D, \text{RECCOL}_{a,b})$ is Σ_3 -complete.

Proof. Let TOT_a be the set of indices for total Turing machines whose image is contained in $\{1, \dots, a\}$. Note that TOT_a is Π_2 . The set A defined below is a Σ_3 solution of $(D, \text{RECCOL}_{a,b})$. A is the set of ordered pairs $\langle e_1, e_2 \rangle$ such that $(e_1 \in \text{TOT01}) \wedge (e_2 \in \text{TOT01})$, and there exists i such that

- (1) $i \in \text{TOT}_a$, and
- (2) $(\forall x, y) [(x \neq y) \wedge (\{e_1\}(x) = \{e_1\}(y) = 1) \\ \wedge ((\{i\}(x) = \{i\}(y)) \Rightarrow \{e_2\}(x, y) = 0)]$

This definition of A can easily be put into Σ_3 form.

We show that $(D, \text{RECCOL}_{a,b})$ is Σ_3 -hard by showing that if A is a solution to $(D, \text{RECCOL}_{a,b})$, then $\text{COF} \leq_m A$. Given x , we construct a recursive graph $G(x) = G$ such that

- (i) $x \in \text{COF} \Rightarrow \chi^r(G) = a$, and
- (ii) $x \notin \text{COF} \Rightarrow \chi^r(G) = b$.

We use a modification of the construction in Theorem 4.15 of a recursive graph G such that $\chi(G) = a$, but $\chi^r(G) = b$. In this modification, we weave the set W_x into the construction in such a way that if W_x is cofinite, then the construction fails and $\chi^r(G) = a$; and if W_x is not cofinite then the construction succeeds and $\chi^r(G) = b$.

Let $\mathbf{N} = \bigcup_{e,i} X_e^i$ be a recursive partition of $\mathbf{N}$ into an infinite number of infinite recursive sets. Let $y_{\langle e,i \rangle}$ be defined such that

$$y_{\langle e,i \rangle} \in \text{TOT} \quad \text{iff} \quad \{\langle e, i \rangle, \langle e, i \rangle + 1, \dots\} \subseteq W_x.$$

Let $G(e, i) = (V(e, i), E(e, i))$ be the recursive graph obtained by applying Lemma 4.19 to a, e, i, X_e^i , and $y_{\langle e, i \rangle}$. Let $G = \bigcup_e \bigcup_{0 \leq i < b} G(e, i)$, and $G = (V, E)$. Clearly G is recursive and $\chi(G) = a$.

If $x \notin \text{COF}$, then for all e, i we have $y_{\langle e, i \rangle} \notin \text{TOT}$. Hence, by Lemma 4.19, for all e and all $i < b$, $G(e, i)$ is not i -colored by $\{e\}$. Therefore $\chi^r(G) \geq b$. By Lemma 4.19 (vi) (c), the graphs $G(e, i)$ are recursively $(i + 1)$ -colorable in a uniform way, hence $\chi^r(G) \leq b$. Combining these two, yields $\chi^r(G) = b$. (Note that this argument holds when $b = \infty$.)

If $x \in \text{COF}$, then

$$S' = \{\langle e, i \rangle \mid y_{\langle e, i \rangle} \notin \text{TOT} \wedge 0 \leq i < b \wedge e \in \mathbb{N}\}$$

is finite. Let

$$S'' = \{\langle e, i \rangle \mid y_{\langle e, i \rangle} \in \text{TOT} \wedge 0 \leq i < b \wedge e \in \mathbb{N}\},$$

$G' = \bigcup_{\langle e, i \rangle \in S'} G(e, i)$ and $G'' = \bigcup_{\langle e, i \rangle \in S''} G(e, i)$. Note that $G = G' \cup G''$. We show that $G' \cup G''$ is recursively a -colorable, by showing that G' and G'' are recursively a -colorable (and using that $G = G' \cup G''$ is a recursive partition of G).

If $\langle e, i \rangle \in S'$ then $y_{\langle e, i \rangle} \notin \text{TOT}$ so, by Lemma 4.19, $G(e, i)$ is finite and $\chi(G(e, i)) = a$. Since S' is finite, G' is a finite a -colorable graph. Hence $\chi^r(G') = a$.

If $\langle e, i \rangle \in S''$ then $y_{\langle e, i \rangle} \in \text{TOT}$ so, by Lemma 4.19, one can effectively find the finite component of $G(e, i)$ in which a given $v \in V(e, i)$ is contained. We use this to recursively a -color G'' . Let $G'' = (V'', E'')$.

Given a number v , first check if it is in V'' . If it is not then output 1 and halt (we need not color it). If $v \in V''$ then run the construction until you find e, i such that $v \in G(e, i)$. Then find the finite component of $G(e, i)$ that contains v . Let c be the least lexicographic coloring of this component. $\square$

4.4 Combinatorial modification

Theorem 4.15 shows that there are recursive graphs G such that, even though $\chi(G) = 2$, no finite number of colors will suffice to color it recursively. If G is highly recursive, then more colors do help. The following theorem was first proven in [145] but also appears in [31] (independently).

Theorem 4.21 *If G is highly recursive and n -colorable, then G is recursively $(2n - 1)$ -colorable. Moreover, given an index for G (as a highly recursive graph), one can recursively find an index for a $(2n - 1)$ -coloring of G .*

Proof. Assume, without loss of generality, that $V = \mathbb{N}$. Color vertex 1 with color 1. Assume the following inductively

- (1) The vertices $\{1, \dots, m\}$ are colored with $\{1, \dots, 2n - 1\}$ (additional vertices may also be colored with $\{1, \dots, 2n - 1\}$, but not with any other colors).
- (2) Let

$$B_m = \{v \mid v \text{ is colored and } v \text{ is adjacent to a vertex that is not colored}\}.$$

The vertices of B_m are colored with either $\{1, \dots, n - 1\}$ or $\{n + 1, \dots, 2n - 1\}$.

We color the vertex $m + 1$ (and possibly some additional vertices). If $m + 1$ is already colored, then note that (1) and (2) hold for $m + 1$, so proceed to color $m + 2$. Otherwise we color $m + 1$ as follows. Let H be the set

$$\{v \mid v \text{ is not colored, } \exists u \text{ that is colored such that } d(u, v) \leq 2\} \cup \{m + 1\}.$$

($d(u, v)$ is the length of the shortest path from u to v .) Assume B_m is colored with $\{1, \dots, n - 1\}$ (the case where B_m is colored with $\{n + 1, \dots, 2n - 1\}$ is similar). Color H with $\{n, \dots, 2n - 1\}$ such that vertex $m + 1$ does not receive color n . We now want to ensure that B_{m+1} is colored with $\{n + 1, \dots, 2n - 1\}$, i.e., does not use color n . This will involve uncoloring some vertices. Let

$$B' = \{v \mid v \text{ is colored } n \text{ and } v \text{ is adjacent to a non-colored vertex}\}.$$

We uncolor all the vertices in B' . Note that $B' \cap \{1, \dots, m + 1\} = \emptyset$, so (1) holds. Note that all colored neighbors of B' use colors $\{n + 1, \dots, 2n - 1\}$, hence all vertices in B_{m+1} use only these colors, so (2) holds. $\square$

We will see later (Theorem 4.30) that the upper bound cannot be improved for general highly recursive graphs. However, if G is connected and $\chi(G) = 2$, then it is easy to see that $\chi^r(G) = 2$. This holds for both G recursive and G highly recursive.

4.5 Recursive analogue is false for highly recursive graphs

Theorem 4.21 gives an upper bound on the number of colors required to color an n -colorable highly recursive graph. The question arises, “Can we do better?”. We cannot! That is, there exists a highly recursive graph G such that $\chi(G) = n$ and $\chi^r(G) = 2n - 1$. The proof requires several definitions and lemmas. It was first proven in [145] but our exposition is based on a modification which was presented in [14].

Definition 4.22 Let $n \geq 3$. Let $G^n = (V, E)$ where

$$V = \{(i, j) : 1 \leq i, j \leq n\},$$

$$E = \{\{(i, j), (r, s)\} : i \neq r \text{ and } j \neq s\}.$$

If $1 \leq i \leq n$, then the set of vertices $\{(i, j) : 1 \leq j \leq n\}$ is called the i -th column of G^n . The j -th row of G^n is defined similarly. The *basic row coloring* of G^n assigns color i to every vertex in the i -th row. The *basic column coloring* of G^n assigns color i to every vertex in the i -th column. Note that both are valid vertex colorings of G^n using only n colors.

Definition 4.23 If χ is a coloring of G^n , then χ induces a *colorful column (row)* if χ assigns to each vertex in a particular column (row) a different color. If the coloring being referred to is obvious, we may say “ G has a colorful column (row)” to mean that the coloring induces a colorful column (row).

Lemma 4.24 Let χ be a coloring of G^n . Let $1 \leq i, j \leq n$ with $i \neq j$.

- (1) If color a appears more than once in row i (column i), then a cannot appear in row j (column j).
- (2) If color a appears more than once in row i , and color b appears more than once in column j , then $a \neq b$.

Proof.

(1) Assume $\chi((x, i)) = \chi((y, i)) = a$. Every vertex (z, j) is connected to either (x, i) , or (y, i) , or both, hence $\chi((z, j)) \neq a$. Similar for the column version.

(2) Assume $\chi((x, i)) = \chi((y, i)) = a$ and $\chi((j, w)) = \chi((j, z)) = b$. Since $x \neq y$, one of $\{x, y\}$ is not j ; assume $x \neq j$. Since $w \neq z$, one of $\{w, z\}$ is not i . If $z \neq i$, then (x, i) and (j, z) are connected, so $a \neq b$. If $w \neq i$, then (x, i) and (j, w) are connected, so $a \neq b$. $\square$

Lemma 4.25 *If χ is a $(2n - 2)$ -coloring of G^n , then χ either induces a colorful row or induces a colorful column, but not both.*

Proof. Assume, by way of contradiction, that χ is a $(2n - 2)$ -coloring of G^n that induces neither a colorful row nor column. For $1 \leq i \leq n$ let a_i (b_i) be a color that appears more than once in row i (column i). By Lemma 4.24,

(1) for all i, j with $i \neq j$, $a_i \neq a_j$ and $b_i \neq b_j$, and

(2) for all i, j , $a_i \neq b_j$.

Hence the set $\{a_1, \dots, a_n, b_1, \dots, b_n\}$ has $2n$ colors, which contradicts χ being a $(2n - 2)$ -coloring. Hence χ induces a colorful row or column.

Assume, by way of contradiction, that χ induces a colorful row and a colorful column. Let row i and column j be colorful. The only way a vertex (k, i) in the i -th row could have the same color as a vertex (j, m) in the j -th column is if they are not connected, i.e., $k \neq j$ or $i \neq m$. If only one of these holds, then (k, i) and (j, m) are in the same row or column, and are colored differently; hence we must have $k = j$ and $i = m$. This means they are the same vertex. This happens exactly once, so the colorful row and column use a total of $2n - 1$ colors. This contradicts χ being a $(2n - 2)$ -coloring. $\square$

We now define a way to connect two graphs such that if in some coloring one of them has a colorful row (column) the other will have a colorful column (row).

Definition 4.26 Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be two graphs such that $G_1 \cong G_2 \cong G^n$. Assume the vertices of G_k are of the form (k, i, j) in such a way that (k, i, j) corresponds to (i, j) . The following graph is the 2-element chain of G_1 and G_2 , denoted $CH(G_1, G_2)$.

$$V = V_1 \cup V_2$$

$$E = E_1 \cup E_2 \cup E_{12}$$

$$E_{12} = \{(1, i, j), (2, r, s)\} : i \neq s \text{ and } r \neq j\}.$$

The edges in E_{12} are said to *link together* G_1 and G_2 . Let $G_1, \dots, G_s$ be graphs of type G^n . The s -element chain of $G_1, \dots, G_s$, denoted by the expression $CH(G_1, \dots, G_s)$, can be defined by linking G_1 to G_2 , G_2 to G_3 , $\dots$, G_{s-1} to G_s .

In $CH(G_1, G_2)$ the r -th row of G_2 acts like the r -th column of G_1 in terms of which vertices of G_1 it is connected to. This intuition underlies the next lemma.

Lemma 4.27 *Let χ be a $2n-2$ partial coloring of $CH(G_1, G_2)$ that induces a colorful row (column) of the G_1 part. Any extension of χ to a $(2n-2)$ -coloring of $CH(G_1, G_2)$ must induce a colorful column (row) in the G_2 part.*

Proof. Let χ and i be such that χ is a $2n-2$ partial coloring of $CH(G_1, G_2)$ that induces the i -th column of G_1 to be colorful. Assume, by way of contradiction, that there exists χ_0 such that χ_0 is a $(2n-2)$ -coloring of $CH(G_1, G_2)$ which is an extension of χ , but does not induce a colorful row of G_2 .

Every row of G_2 has some color repeated at least twice. For $1 \leq j \leq n$, let $c_j = \chi_0((1, i, j))$. For $1 \leq s \leq n$, let d_s be the color that, using χ , appears twice in the s -th row of G_2 .

We show that $|\{c_1, \dots, c_n, d_1, \dots, d_n\}| > 2n-2$. Since the i -th row was colorful, all the c_j 's are distinct. By Lemma 4.24 all the d_s 's are distinct. To show $|\{c_1, \dots, c_n, d_1, \dots, d_n\}| > 2n-2$, we show that the only element that may be in $\{c_1, \dots, c_n\} \cap \{d_1, \dots, d_n\}$ is d_i .

Assume $c_j = d_s$. Let r_1 and r_2 be such that $\chi((2, r_1, s)) = \chi((2, r_2, s)) = d_s$. By definition, $\chi((1, i, j)) = c_j$. Since $c_j = d_s$, there is no edge between $(2, r_1, s)$ and $(1, i, j)$; hence either $r_1 = j$ or $s = i$. Similarly, there is no edge between $(2, r_2, s)$ and $(1, i, j)$; hence either $r_2 = j$ or $s = i$. If $s \neq i$, then $r_1 = j$ and $r_2 = j$; so $r_1 = r_2$, which is false. Hence $s = i$. Therefore the only element of $\{c_1, \dots, c_n\} \cap \{d_1, \dots, d_n\}$ is d_i . $\square$

Lemma 4.28 *Let χ be a $2n-2$ partial coloring of $CH(G_1, \dots, G_s)$ that induces a colorful row (column) of the G_1 part. If s is even, then any extension χ_0 of χ to a $(2n-2)$ -coloring of $CH(G_1, \dots, G_s)$ must induce a colorful column (row) of the G_s part. If s is odd, then χ_0 must induce a colorful row (column).*

Proof. This follows from the previous lemma and induction. $\square$

Lemma 4.29 *The graph $CH(G_1, \dots, G_s)$ is n -colorable.*

Proof. For i even, color G_i by coloring every vertex in row j with color j . For i odd, color G_i by coloring every vertex in column j with color j . This is easily seen to be an n -coloring of $CH(G_1, \dots, G_s)$. $\square$

Theorem 4.30 *Let $n \geq 3$. There exists a highly recursive graph $\tilde{G}$ such that $\chi(\tilde{G}) = n$ and $\chi^r(\tilde{G}) = 2n - 1$.*

Proof. We construct a highly recursive graph $\tilde{G}$ to satisfy the following requirements:

R_e : If $\{e\}$ is total, then $\{e\}$ is not a $(2n - 2)$ -coloring of $\tilde{G}$.

Recursively partition $\mathbb{N}$ into infinite sets $X_0, X_1, X_2, \dots$. We satisfy R_e using vertices from X_e . Fix e . We show how to construct a highly recursive graph $G = G(e)$ such that

(1) $\chi(G) = n$, and

(2) $\{e\}$ is not a $(2n - 2)$ -coloring of G .

The graph $\tilde{G}$ is then simply $\bigcup_{e=0}^{\infty} G(e)$.

We construct $G = G(e)$ in stages. To avoid confusion we **do not** use “ G_s ”, we merely speak of “ G at stage s ”.

Construction of $G = G(e)$

STAGE 0:

At this stage, G consists of two graphs G_1 and G_2 , with $G_1 \cong G_2 \cong G^n$.

STAGE $s + 1$:

(At the end of stage s , G consists of $CH(G_1, G_3, \dots, G_{2s+1})$ and $CH(G_2, G_4, \dots, G_{2s+2})$, where each G_i is isomorphic to G^n .) Run $\{e\}_s$ on all the vertices of G_1 and G_2 . There are several cases.

Case 1. There exists a vertex in G_1 or G_2 where $\{e\}_s$ does not converge. Let G_{2s+3} and G_{2s+4} be graphs isomorphic to G^n that use the least numbers from X_e that are larger than s , but are not already in G , for vertices. Extend the $(s + 1)$ -chains to $(s + 2)$ -chains using G_{2s+3} for the odd chain, and G_{2s+4} for the even chain.

Case 2. $\{e\}_s$ converges on all the vertices of G_1 and G_2 , and either uses more than $2n - 2$ colors, or is not a coloring. Stop the construction of G since R_e is satisfied.

Case 3. $\{e\}_s$ converges on all the vertices in G_1 and G_2 , uses at most $2n - 2$ colors, is a coloring, and both G_1 and G_2 have colorful rows (columns). By Lemma 4.28, any extension of $\{e\}_s$ to a coloring of G will induce G_{2s+1} and G_{2s+2} to either both have a colorful column or both have a colorful row. We link G_{2s+1} to a (new) graph $H \cong G^n$, and then link G_{2s+2} to H (all new vertices are the least unused vertices of X_e). By Lemma 4.27, any extension of the coloring must induce both a colorful row and a colorful column in H . By Lemma 4.25, H cannot be $2n - 2$ colored in this manner. Stop the construction, as R_e is satisfied.

Case 4. $\{e\}_s$ converges on all the vertices in G_1 and G_2 , and is a $(2n - 2)$ -coloring of both G_1 and G_2 ; and this coloring induces G_1 to have a colorful row (column), and G_2 to have a colorful column (row). Link both G_{2s+1} and G_{2s+2} to a graph isomorphic to G^n . The coloring $\{e\}_s$ cannot be extended to a $(2n - 2)$ -coloring of G , since in such a coloring the new G^n graph would have to have both a colorful row and a colorful column. End of the construction.

We show that the graph $G(e)$ is highly recursive. The vertex set is recursive since v is a vertex iff $v \in X_e$ and v was placed into the graph at some stage $s \leq v$. To determine the neighbors of a vertex v note that, if v enters the graph at stage s , then all the neighbors of v will enter by stage $s + 1$.

Since $G(e)$ is highly recursive, it follows from Theorem 4.21 that $\chi^r(G) \leq 2n - 1$ (this can also be proven directly). By the comments made before and during the construction, it is easy to see that $\tilde{G}$ (the union of all the G 's) is not recursively $(2n - 2)$ -colorable. Hence $\chi^r(\tilde{G}) = 2n - 1$.

By Lemma 4.29 $\chi(\tilde{G}) = n$. □

4.6 Recursion-theoretic modification

By Theorems 4.15 and 4.30, n -colorable recursive and highly recursive graphs need not be recursively n -colorable. But such graphs do have n -colorings of low degree.

The following theorem is due to Bean [11].

Theorem 4.31 *If G is a recursive graph that is n -colorable, then there exists an n -coloring c that is of low degree.*

Proof. Let G be as in the hypothesis. Consider the following recursive n -ary tree T . The vertex $\sigma = (a_1, \dots, a_m)$ (where for all i , $1 \leq a_i \leq n$) is on the tree T iff it represents a non-contradictory coloring (i.e., the partial coloring of G that colors vertex i with a_i does not label two adjacent vertices with the same color). We have

- (1) T is recursive,
- (2) T is recursively bounded by the function $f(m) = \langle n, \dots, n \rangle$ (the n 's appear m times),
- (3) any infinite branch of T is an n -coloring of G , and
- (4) every n -coloring of G is represented by some infinite branch of T (note that the set of infinite branches is nonempty, since G is n -colorable).

Since the infinite branches of T form a nonempty recursively bounded Π_1^0 class, by Theorem 2.12 there exists an infinite low branch. Since every infinite branch is an n -coloring, the theorem follows. $\square$

The proof of the above theorem actually shows that the set of n -colorings of G is a recursively bounded Π_1^0 class. Remmel [139] has shown the converse: for any recursively bounded Π_1^0 class $\mathcal{C}$ and any constant $n \geq 3$, there exists a highly recursive graph G such that (up to a permutation of colors) there is an effective one-to-one degree-preserving correspondence between n -colorings of G and elements of $\mathcal{C}$.

4.7 Miscellaneous

We state several results about recursive-graph colorings without proof.

4.7.1 Bounding the genus

If the genus of a (finite or infinite) graph G is bounded by g , then $\chi(G)$ is bounded by a function of g which we denote $c(g)$. In 1890, Heawood [78]

showed that, for $g \geq 1$,

$$c(g) \leq \left\lfloor \frac{7 + \sqrt{48g + 1}}{2} \right\rfloor.$$

In 1967, Ringel and Young [140] proved that this bound is tight. (See Chapter 5 of [69] for proofs of both the upper and lower bound). Appel and Haken [4, 5, 6] showed that $c(0) = 4$ (e.g., planar graphs are 4-colorable) using very different techniques which involved a rather long computer search.

We wonder if an analogue of $c(g)$ exists for recursive or highly recursive graphs (i.e., perhaps every recursive graph of genus g is $c'(g)$ -colorable for some c'). Since the graph G constructed in Corollary 4.16 is planar (i.e., its genus is 0) and $\chi^r(G) = \infty$, no analogue of $c(g)$ exists for recursive colorings of recursive graphs. However, for highly recursive graphs G , Bean [11] showed that $\chi^r(G) \leq 2(c(g) - 1)$. Using the 4-color Theorem this yields that, if G is planar, then $\chi^r(G) \leq 6$ (Bean [11] obtained this result without using the 4-color Theorem). Using the 4-color Theorem, Carstens [27, 29] claims to have shown¹ that, if G is a highly recursive planar graph, then $\chi^r(G) \leq 5$. It is an open problem to obtain $\chi^r(G) \leq 4$. More generally, it is an open problem to improve Bean's bound for general genus g , or show it cannot be improved. It may be of interest to impose additional recursion-theoretic constraints such as having a recursive embedding on a surface of genus g .

4.7.2 Bounding the degree

The *degree* of a graph is the maximal degree of a vertex. A graph G satisfies Δ_d if it has degree d and does not have a subgraph isomorphic to K_{d+1} (the complete graph on $d + 1$ vertices).

Brooks [22] showed that if a (finite or infinite) graph G satisfies Δ_d then $\chi(G) \leq d$. By a variation of Theorem 4.15, for every d there exists a recursive graph G that satisfies Δ_d but $\chi^r(G) = d + 1$; hence the recursive analogue of Brooks's theorem fails for recursive graphs. Schmerl [146] showed that the recursive analogue does hold for highly recursive graphs. Carstens and Pappinghaus[31] discovered the result independently, and Tverberg [164] has given a simpler proof. It is an open question as to just how wide the gap between $\chi(G)$ and $\chi^r(G)$ may be for recursive graphs with property Δ_d .

¹The paper sketches the proof and promises further work with details that, to our knowledge, has not appeared.

4.7.3 Regular graphs

A graph is d -regular if every vertex has degree d . Note that a recursive d -regular graph is highly recursive. Schmerl [145] posed the following question: “If $2 \leq n \leq m \leq 2n - 2$, what is the least d for which there is a recursive d -regular graph which is not recursively m -colorable?”. We denote this quantity by $d(n, m)$.

Schmerl [145] notes that if there exists a highly recursive G with degree bound d , then there exists a highly recursive d -regular G' such that $\chi(G) = \chi(G')$ and $\chi^r(G) = \chi^r(G')$. Bean constructed, for every $k \geq 2$, a highly recursive G , with $\chi(G) = k$, $\chi^r(G) = k + 1$, and degree bound $2k - 2$. Hence $d(n, n + 1) \leq 2n - 2$ (Manaster and Rosenstein [120] obtained the same result with different methods). Schmerl showed how to modify Bean’s construction to obtain

$$d(n, n + 1) \leq \left\lfloor \frac{3n - 1}{2} \right\rfloor.$$

Since the degree bound of the graph constructed in Theorem 4.30 (of this survey) is $3(n - 1)^2$,

$$d(n, 2n - 2) \leq 3(n - 1)^2.$$

4.7.4 Perfect graphs

It is of interest to impose graph-theoretic conditions on a highly recursive graph G such that if G satisfies the condition then $\chi^r(G)$ is not too far from $\chi(G)$. A graph is *perfect* if for every induced subgraph H , $\omega(H) = \chi(H)$ (where $\omega(H)$ is the size of the largest clique in H). Kierstead [95] proved that, if G is a highly recursive perfect graph, then $\chi^r(G) \leq \chi(G) + 1$. It is a (vague) open question to find other graph-theoretic conditions that narrow the gap between $\chi(G)$ and $\chi^r(G)$.

4.7.5 On-line colorings

Informally, an on-line algorithm to color an (infinite) graph is an algorithm that is given the graph a vertex at a time, and has to color a vertex as soon as it sees it. For general graphs it is hopeless to try to bound the number of colors such an algorithm will need to use, but we can bound the number of colors it uses on the first n vertices. It is trivial to color the first n vertices with n colors. Lovász, Saks, and Trotter [117] have improved this by showing

- (i) if $\chi(G) \leq 2$, then G can be colored on-line via an algorithm that uses $2 \log n$ colors on the first n vertices;
- (ii) if $\chi(G) \leq k$ then G can be colored on-line via an algorithm that uses

$$O\left(n \frac{\log^{(2k-3)} n}{\log^{(2k-4)} n}\right)$$

colors on the first n vertices ($\log^{(m)}$ n is iterated $\log$).

There are limits on the extent to which these bounds can be improved: Vishwanathan [165] showed that for every on-line algorithm, and every k and n , there exists a graph G on n vertices (and an order to present it) so that $\chi(G) \leq k$, and the algorithm must use at least $(\frac{\log n}{4k})^{k-1}$ colors. Irani [86] has shown that certain classes of graphs (which include planar graphs) have a presentation with which they are on-line colorable with $O(\log n)$ colors. For a survey of this area see [98, 103].

There are connections between on-line coloring algorithms and combinatorial analogues of Dilworth's Theorem. See Section 6.4.2 for an overview.

4.7.6 Coloring directed graphs

More complex conditions can be imposed on directed graphs than on undirected graphs. Kierstead [97] has found one such condition that affects the recursive chromatic number. He has shown that if G is a recursive directed graph that does not have an induced subgraph of the form

- (i) directed 3-cycle, or
- (ii) $\circ \longrightarrow \circ \longrightarrow \circ \longleftarrow \circ$, or
- (iii) $\circ \longleftarrow \circ \longrightarrow \circ \longrightarrow \circ$,

then $\chi^r(G) \leq 2^{\omega(G)}$, where $\omega(G)$ is the size of the largest clique.

4.7.7 Coloring interval graphs

An *interval graph* is the comparison graph for an interval order (see Section 6.7.1 for the definition of an interval order). Kierstead and Trotter [102] have shown that recursive interval graphs are $(3\omega(G) - 2)$ -colorable, where $\omega(G)$ is the size of the largest clique in G .

4.7.8 Decidable graphs

Bean [11] considered imposing stronger recursive conditions on a graph than highly recursive.

Definition 4.32 A graph G is *decidable* if there is a decision procedure to determine if a given first order sentence about it is true. The language in which the sentences are expressed has

- (1) the usual logic symbols including quantifiers that range over vertices,
- (2) the symbol $E(x, y)$ (tests if x and y are connected by an edge), and the symbol “=” for equality.

Bean showed that all negative results that he obtained for highly recursive graphs also hold for decidable graphs. Combining his technique with Theorem 4.30 yields that for every $k \geq 2$ there exists a decidable graph G such that $\chi(G) = k$ and $\chi^r(G) = 2k - 1$. That construction can be combined with the recursion-theoretic techniques of Theorem 4.20 to obtain that the set of indices of decidable graphs G such that $\chi(G) = k$ and $\chi^r(G) = 2k - 1$ is Σ_3 -complete.

It is an open question to find a reasonable recursive condition for graphs G that implies $\chi(G) = \chi^r(G)$. Expanding the language in which the sentences are expressed may help. A comprehensive study of types of decidable graphs has not been undertaken.

Dekker[46] examined graphs where one can decide whether two vertices are connected by a path, but he did not examine coloring.

4.7.9 A -recursive graphs

Gasarch and Lee [65] considered graphs that were intermediary between recursive and highly recursive. Let nbd_G be the function that, on inputting x (a vertex of G) outputs all the neighbors of G . Note that if G is recursive then $nbd_G \leq_T K$, and if G is highly recursive then $nbd_G \leq_T A$.

Definition 4.33 Let A be any set. A graph $G = (V, E)$ is *A -recursive* if G is recursive and $nbd(G) \leq_T A$.

A natural question is to see, for various sets A , if G being A -recursive implies any finite bound on $\chi^r(G)$. The answer is no:

Theorem 4.34 *Let A be a non-recursive r.e. set. Then there exists an A -recursive graph G such that G is 2-colorable but not recursively k -colorable for any natural number k .*

The proof is a variant of Bean's original construction with a permitting argument. The former enables us to show that the graph is 2-colorable but not recursively k -colorable for any k . The latter allows us to show that the neighbor function is recursive in the r.e. set A . For a discussion of the permitting method, see [159]. It is an open question to extend the theorem to all A such that $\emptyset <_T A <_T K$. Even the case where A is 2-r.e. is open.

The proof technique can be extended to show the following generalization of Theorem 4.15.

Theorem 4.35 *Let A be a non-recursive r.e. set. Let a, b be such that $2 \leq a < b \leq \infty$. Let X be an infinite recursive set. There exists an A -recursive graph $G = (V, E)$ such that $\chi(G) = a$, $\chi^r(G) = b$, and $V \subseteq X$. If $a \leq 4$, then G can be taken to be planar.*

4.7.10 Complexity of finding $\chi(G)$ and $\chi^r(G)$

Theorem 4.20 says that determining $\chi^r(G)$ will require a $\emptyset'''$ oracle. A comprehensive study of *how many queries* are required to determine $\chi(G)$ and $\chi^r(G)$ was undertaken by Beigel and Gasarch [14, 15]. In those papers 64 questions were raised (six 2-valued parameters were varied), of which 58 were solved exactly. We present two theorems that encompass four of these questions.

Theorem 4.36 *Let $c \geq 2$. Let D_c (D_c^r) be the set of indices of recursive graphs G such that $\chi(G) \leq c$ ($\chi^r(G) \leq c$). Let χ_c and χ_c^r be the partial functions*

$$\chi_c(e) = \begin{cases} \chi(G_e^r) & \text{if } e \in D_c, \\ \uparrow & \text{if } e \notin D_c. \end{cases} \quad \text{and} \quad \chi_c^r(e) = \begin{cases} \chi^r(G_e^r) & \text{if } e \in D_c^r, \\ \uparrow & \text{if } e \notin D_c^r. \end{cases}$$

There is a solution to the promise problem (D_c, χ_c) that can be computed with $\lceil \log(c+1) \rceil$ queries to K . For every set X , no solution to (D_c, χ_c) can be computed with $\lceil \log(c+1) \rceil - 1$ queries to X . If X is any set such that $K \not\leq_T X$ then (D_c, χ_c) cannot be computed with X . Similar theorems hold for computing (D_c^r, χ_c^r) with oracle $\emptyset'''$. Similar theorems hold for highly recursive graphs.

Theorem 4.37 *Let f and g be recursive functions such that*

- (1) $\sum_{i=0}^{\infty} 2^{-f(i)} \leq 1$, *and is an effectively computable real r (i.e., there exists a recursive function $h : \mathbb{Q} \rightarrow \mathbb{Q}$ such that $|h(\epsilon) - r| < \epsilon$), and*
- (2) $\sum_{i=0}^{\infty} 2^{-g(i)} > 1$.

Let D be the set of valid indices for recursive graphs. There is a solution for the promise problem (D, χ) that, on inputting e , takes $f(\chi(G_e))$ queries to K . For every set X , no solution to (D, χ) can be computed with $g(\chi(G_e))$ queries to X . Similar results hold for χ^r with a ϕ''' oracle. Similar results hold for highly recursive graphs.

4.7.11 Actually finding a coloring

None of the results looked at so far involve actually coloring the graph. Beigel and Gasarch [17] examined this issue in terms of the number of times a recursive procedure will have to change its mind while coloring a graph. They constructed graphs where a recursive mapmaker has to recolor the map many times.

Definition 4.38 Let $G = (V, E)$ be a k -colorable recursive graph. A *local k -coloring* of G is a function that takes a finite set $H \subseteq V$ and outputs a k -coloring of H that is extendible to a k -coloring of all of G .

We examine the complexity of local k -colorings. Our measure of complexity is “mind-changes”. In particular, we study algorithms for local k -colorings that are allowed to change their mind $g(n)$ times on inputs consisting of n vertices. The function g is the complexity of the algorithm. There are recursive graphs for which every local coloring changes its mind many times.

In what follows we will interpret the input to a Turing machine as an ordered pair (H, s) where H is a finite set of vertices and s is a parameter; and the output as a coloring of those vertices.

Definition 4.39 Let f be a function from $[\mathbb{N}]^{<\omega}$ to $\mathbb{N}$, and let g be a function from $\mathbb{N}$ to $\mathbb{N}$. The function f is *computable by a g -mind-change algorithm* if there exists a total Turing machine M such that, for every $H \in [\mathbb{N}]^n$

- (1) $\lim_{s \rightarrow \infty} M(H, s) = f(H)$, i.e., $(\exists s_0)(\forall s \geq s_0) M(H, s) = f(H)$, and
- (2) $|\{s : M(H, s) \neq M(H, s+1)\}| \leq g(n)$.

Carstens and Pappinghaus [31] showed that one can color any recursive graph with a mind-change algorithm that changes its mind an exponential number of times. We sharpen their result and put it in our terminology. Let $NI(n, k)$ be the number of non-isomorphic colorings of n vertices with k colors. It can be shown that

$$NI(n, k) = \sum_{t=0}^k \frac{t^n}{t!} \sum_{r=0}^{k-t} \frac{(-1)^r}{r!}.$$

For large n and fixed k this is approximately $k^n/k!$.

Theorem 4.40 *Let $k \geq 3$. Let $G = (V, E)$ be a k -colorable recursive graph. There exists a local k -coloring of G that is computable by a g -mind-change algorithm, where $g(n) = NI(n, k) - 1$. There exists a k -colorable recursive graph G such that every mind-change algorithm that computes a local k -coloring of G requires $NI(n, k) - 1$ mind-changes on an infinite number of inputs H of arbitrarily large cardinality.*

4.7.12 Polynomial graphs

Cenzer and Remmel [35] have considered graphs with labels in $\{0, 1\}^*$, such that testing for an edge can be done in polynomial time. They have shown the following.

Theorem 4.41

- (i) *If G is a recursive graph and $k \in \mathbb{N}$, then there exists a poly graph G' such that there is an effective degree-preserving map from the k -colorings of G to the k -colorings of G' . Hence, using Theorem 4.15, there exists a poly graph that is 3-colorable, but not recursively k -colorable for any k .*
- (ii) *There exists a poly graph G that is 2-colorable, connected, but not primitive-recursively 2-colorable. This is of interest since it shows that the natural analog of Note 4.17 (ii) is false.*

5 Hall's Theorem on bipartite graphs

We consider the infinite version of Hall's Theorem on solutions to bipartite graphs. We present the finite and infinite versions (due to Phillip Hall [75] and Marshall Hall [74] respectively), show that a recursive analogue of Hall's Theorem is false, and show that a recursion-theoretic modification is true. We will then show that there is a modification that is both recursion-theoretic and combinatorial which is true, and finally state some miscellaneous results.

Hall's theorem for finite graphs also yields an algorithm for testing if a bipartite graph has a solution, and, if so, finding it. These algorithms are not efficient. See [131] or [58] for efficient algorithms for these problems.

5.1 Definitions and classical version

Definition 5.1 A *bipartite graph* G is a 3-tuple (A, B, E) where A and B are disjoint sets of vertices, and $E \subseteq [A \cup B]^2 - ([A]^2 \cup [B]^2)$ (i.e., E consists of unordered pairs of vertices, one from A and one from B). If $\forall x \in A \cup B$, $\text{degree}(x) < \infty$, then we say G has *finite degree*. Henceforth G has finite degree. The neighbors of a finite set of vertices $X \subseteq A$ are denoted $nb_G(X)$. Formally we define $nb_G : P^{<\omega}(A) \rightarrow P^{<\omega}(B)$ as follows: for each finite $X \subseteq A$,

$$nb_G(X) = \{b \in B \mid (\exists a \in X) \{a, b\} \in E\}.$$

Note that from the function nb_G one can obtain all the edges of G . When G is clear from context we abbreviate nb_G by nb .

Definition 5.2 Let $G = (A, B, E)$ be a bipartite graph. A function $f : A \rightarrow B$ is a *solution* for G if f is one-to-one and $\forall a \in A \{a, f(a)\} \in E$. Given $X \subseteq A$ and $Y \subseteq B$, we will sometimes call $f : X \rightarrow Y$ a *solution from X to Y* . If f is onto then the solution is *symmetric*.

We will be considering the infinite version of Hall's Theorem. We present the finite and infinite versions. The proof we give for the finite case does not lead to a computationally efficient algorithm to find a solution. The most efficient algorithm known for this problem runs in time $O(|V|^{\frac{1}{2}}|E|^2)$ (see [131, p. 226]).

Definition 5.3 Let $G = (A, B, E)$ be a (finite or infinite) bipartite graph. G satisfies *Hall's condition* if for all finite $X \subseteq A$, $|nb_G(X)| \geq |X|$.

Theorem 5.4 (Finite Hall's Theorem) *Suppose $G = (A, B, E)$ is a finite bipartite graph. Then G has a solution iff G satisfies Hall's condition.*

Proof. If G does not satisfy Hall's condition, then there is an $X \subseteq A$ such that $|nb_G(X)| < |X|$. Obviously, there is no solution for X , so there can be no solution for G .

Now suppose G satisfies $\forall X \subseteq A, |X| \leq |nb_G(X)|$. Let $n = |A|$. We will prove by induction on n that there is a solution for G .

If $n = 1$, let $A = \{a\}$. Since $|nb_G(\{a\})| \geq 1$, there is $b \in B$ such that $\{a, b\} \in E$. Then $M = \{(a, b)\}$ is a solution for G .

If $n > 1$, assume the theorem holds for bipartite graphs (A, B, E) with $|A| < n$. We will consider two cases:

Case 1. Suppose for all k with $1 \leq k < n$, and for all $X \subseteq A$ with $|X| = k$, $|nb_G(X)| \geq k + 1$. Then choose any $\{a, b\} \in E$ with $a \in A$. Let G' be $\langle A - \{a\}, B - \{b\}, E' \rangle$, where E' is E restricted to edges that do not involve a or b . Note that for all finite subsets $X \subseteq A - \{a\}$ with $|X| = k$, we have $|nb_{G'}(X)| \geq k$. By our induction hypothesis G' has a solution M . Then $M \cup (a, b)$ is a solution for G .

Case 2. Suppose there is an $X \subset A$ and $k < n$ such that $|X| = |nb_G(X)| = k$. Let $G' = \langle X, B, E' \rangle$, where E' is the set of edges between elements of X and elements of B . By our induction hypothesis, G' has a solution M . Since $|X| = |nb_G(X)|$ and solutions are one-to-one, $image(M) = nb_G(X)$.

Now we need to show there is a solution from $A - X$ to $B - nb_G(X)$. Let $G'' = \langle A - X, B - nb_G(X), E'' \rangle$, where E'' is the subset of E consisting of pairs of elements $\{x, y\}$ such that $x \in A - X$, $y \in B - nb_G(X)$. Assume, by way of contradiction, that there exists $C \subseteq A - X$ such that $|nb_{G''}(C)| < |C|$. Then

$$\begin{aligned} nb_G(C) &= (nb_G(C) \cap nb_G(X)) \cup (nb_G(C) \cap \overline{nb_G(X)}) \\ &= (nb_G(C) \cap nb_G(X)) \cup nb_{G''}(C). \end{aligned}$$

Hence

$$nb_G(C \cup X) = nb_G(C) \cup nb_G(X) = nb_{G''}(C) \cup nb_G(X).$$

We can now show that $|nb_G(C \cup X)| < |C \cup X|$, which contradicts that G satisfies Hall's condition.

$$\begin{aligned} |nb_G(C \cup X)| &= |nb_{G''}(C)| + |nb_G(X)| \\ &= |nb_{G''}(C)| + |X| < |C| + |X| = |C \cup X|, \end{aligned}$$

which contradicts our hypothesis.

Then by our induction hypothesis, there is a solution M' from $A - X$ to $B - nb_G(X)$. Then $M \cup M'$ is a solution for G . $\square$

We now prove the infinite Hall's Theorem. We give a direct proof; it can also be proven by König's Lemma (Theorem 2.3).

Theorem 5.5 (Infinite Hall's Theorem) *Suppose $G = (A, B, E)$ is a countable bipartite graph with finite degree. Then G has a solution iff G satisfies Hall's condition.*

Proof. If G does not satisfy Hall's condition, then there is some finite $X \subseteq A$ such that $|nb_G(X)| < |X|$. Obviously, there is no solution for X , so there can be no solution for G .

Now suppose G satisfies Hall's condition. Since A is countable, let $A = \{a_1 < a_2 < \dots\}$. Given $n \in \mathbb{N}$, let $A^n = \{a_0, \dots, a_n\}$, $B^n = nb_G(A^n)$, $E^n = E \cap \{\{a, b\} \mid a \in A^n, b \in B^n\}$, and $G^n = (A^n, B^n, E^n)$. For $n \in \mathbb{N}$, G^n satisfies Hall's condition, so by the finite Hall's Theorem, there is a solution M^n for G^n . We will build a solution for G from the M^n . Let

$$M(a_1) = \mu x [(\exists^\infty s) M^s(a_1) = x]$$

$$M(a_{n+1}) = \mu x [(\exists^\infty s) (\bigwedge_{j=1}^n (M^s(a_j) = M(a_j)) \wedge M^s(a_{n+1}) = x)]$$

It is easy to see that M is a solution. $\square$

The proof of Theorem 5.5 given above is noneffective. To see if the proof could have been made effective we will look at a potential analogue. In order to state this analogue we need some definitions.

Definition 5.6 Let $G = (A, B, E)$ be a bipartite graph with $A, B \subseteq \mathbb{N}$. G is a *recursive bipartite graph* if A , B and E are recursive, and G has finite degree. Note that a recursive bipartite graph is different from a bipartite recursive graph. G is a *highly recursive bipartite graph* if G is recursive and the function nb_G is recursive. Note that a highly recursive bipartite graph is different from a bipartite highly recursive graph.

We will use the recursive tripling function to represent recursive and highly recursive bipartite graphs.

Definition 5.7 A number $e = \langle e_1, e_2, e_3 \rangle$ determines a *recursive bipartite graph* if $e_1, e_2 \in \text{TOT01}$, $e_3 \in \text{TOT}$, and the sets $A = \{a \mid \{e_1\}(a) = 1\}$ and $B = \{b \mid \{e_2\}(b) = 1\}$ are disjoint. The *recursive bipartite graph determined by e* is (A, B, E) , where

$$E = \{\{a, b\} \mid a \in A, b \in B, \{e_3\}(a, b) = \{e_3\}(b, a) = 1\}.$$

Definition 5.8 A number $e = \langle e_1, e_2, e_3 \rangle$ determines a *highly recursive bipartite graph* if $e_1, e_2 \in \text{TOT01}$, $e_3 \in \text{TOT}$, and the sets $A = \{a \mid \{e_1\}(a) = 1\}$ and $B = \{b \mid \{e_2\}(b) = 1\}$ are disjoint. The *recursive bipartite graph determined by e* is (A, B, E) , where E is determined by the fact that $\{e_3\}$ computes nb_G . (Here, $\{e_3\}$ is interpreted as a function from $\mathbb{N}$ to finite subsets of $\mathbb{N}$.)

Definition 5.9 Let $G = (A, B, E)$ be a bipartite graph such that $A, B \subseteq \mathbb{N}$ (in practice G will be a recursive or highly recursive bipartite graph). A function $f : \mathbb{N} \rightarrow B$ is a *recursive solution* for G if f is total recursive and f , when restricted to A , is a solution for G .

Potential Analogue 5.10 There is a recursive algorithm $\mathcal{A}$ that performs the following. Given an index e for a highly recursive bipartite graph that satisfies Hall's condition, $\mathcal{A}$ outputs an index for a recursive solution. A consequence is that, if a highly recursive bipartite graph G has a solution, then G has a recursive solution.

We will soon see (Theorem 5.13) that this potential analogue is false. We will then have a recursion-theoretic modification which is true. No combinatorial analogue appears to be true; however, we will then impose combinatorial *and* recursion-theoretic conditions that will yield a true analogue.

5.2 Recursive Analogue is False

The following theorem is due to Manaster and Rosenstein [119].

Definition 5.11 If $v_1, v_2, \dots, v_n \in \mathbb{N}$, and all the v_i are distinct, then *the line graph of $(v_1, \dots, v_n)$* is the graph $G = (V, E)$, where $V = \{v_1, \dots, v_n\}$ and $E = \{\{v_i, v_{i+1}\} : 1 \leq i \leq n-1\}$. The vertices $\{v_1, \dots, v_i\}$ are *strictly to the left of v_{i+1}* . The vertex v_1 is the *left endpoint of G* . Terms using ‘right’ instead of ‘left’ can be defined similarly.

We would like to interpret line graphs as bipartite graphs. To do this we need to specify one vertex as being in A (or B), which will determine the status of the other vertices.

The following lemma is easy to prove, hence we leave it to the reader.

Lemma 5.12 *Let $G[i, j]$ be the line graph on*

$$(v_i, v_{i-1}, \dots, v_1, x, y, z, w_1, w_2, \dots, w_j).$$

Interpret $G[i, j]$ as being bipartite by assuming $y \in A$.

- (a) *If i is odd, then (y, x) cannot be in any solution of A to B in G .*
- (b) *If j is odd, then (y, z) cannot be in any solution of A to B in G .*

Proof. We prove *a*; the proof for *b* is similar. We use induction on i .

Let $i = 1$. Note that $v_1 \in A$. If M is any solution of $G[1, j]$ then M must use (v_1, x) , else v_1 (which is in A) will be unmatched. Hence M cannot use (y, x) .

We assume the statement true for odd i and we prove it for $i + 2$. Note that $v_{i+2} \in A$. Let M be a solution of $G[i + 2, j]$. M must use (v_{i+2}, v_{i+1}) , else v_{i+2} will be unmatched. Hence M cannot use (v_i, v_{i+1}) . Therefore $M - \{(v_{i+2}, v_{i+1})\}$ is a solution of $G[i, j]$. By the induction hypothesis, M does not contain (y, x) . $\square$

Theorem 5.13 (Manaster and Rosenstein [119]) *There exists a highly recursive bipartite graph $G = (A, B, E)$ that satisfies Hall’s condition, but has no recursive solution.*

Proof. We construct a highly recursive bipartite graph $G = (A, B, E)$ that has a solution (hence satisfies Hall's condition) and satisfies the following requirements.

R_e : $\{e\}$ total $\Rightarrow \{e\}$ is not a solution of G .

Recursively partition $\mathbb{N}$ into infinite recursive sets $X_0, X_1, \dots$. We construct a highly recursive bipartite graph $G(e) = (A(e), B(e), E(e))$ such that the following hold.

- (i) $A(e) \cup B(e) = X_e$,
- (ii) $G(e)$ has a solution, and
- (iii) $\{e\}$ is not a solution of $G(e)$.

The union $G = \bigcup_{e \geq 0} G(e)$ is the desired graph.

In our description of $G(e)$, whenever we need a vertex, we take the least unused vertex of X_e . We denote the bipartite graph constructed by the end of stage e by $G(e, s) = (A(e, s), B(e, s), E(e, s))$.

Construction

STAGE 0:

Let $G(e, 0)$ be the line graph on (a, b, c) (three new vertices — the least three elements of X_e), interpreted as a bipartite graph by specifying $b \in A(e, 0)$.

STAGE $s + 1$:

(Assume inductively that $G(e, s)$ is a line graph.) Run $\{e\}_s(b)$. There are four cases.

Case 1. If $\{e\}_s(b) \uparrow$ or $\{e\}_s(b) \downarrow \notin \{a, c\}$, then form $G(e, s + 1)$ by adding one vertex to each end of $G(e, s)$.

Case 2. If $\{e\}_{s-1}(b) \uparrow$ and $\{e\}_s(b) \downarrow = a$, then perform whichever of the following two cases applies. In all future stages t never place a vertex on the left end of $G(e, t)$ again.

- (a) if there is an even number of vertices strictly to the left of a in $G(e, s)$, then $G(e, s + 1)$ is formed by placing one vertex on each end of $G(e, s)$;

- (b) if there is an odd number of vertices strictly to the left of a in $G(e, s)$, then $G(e, s + 1)$ is formed by placing one vertex on the right end of $G(e, s)$.

Case 3. If $\{e\}_{s-1}(b) \uparrow$ and $\{e\}_s(b) \downarrow = c$, then this is similar to case 2 except that we are concerned with the right side of $G(e, s)$ and $G(e, t)$.

Case 4. If $\{e\}_{s-1}(b) \downarrow \in \{a, c\}$, then at a previous stage case 2 or 3 must have taken place. $G(e, s + 1)$ is formed by adding a vertex to whichever end of $G(e, s)$ is permitted. End of the construction.

$A(e)$ and $B(e)$ are both recursive: to determine if $p \in A(e)$ or $p \in B(e)$ either

- (1) $p \notin X_e$, so $p \notin A(e)$ and $p \notin B(e)$, or
- (2) $p \in X_e$, in which case run the construction until p appears in the graph, and note whether p enters in the A or B side.

$G(e)$ is highly recursive: if $p \in A(e) \cup B(e)$, then all the neighbors of p appear the stage after p itself appears.

Since $G(e)$ is just the 2-way or 1-way infinite line graph, it obviously has a solution.

We show that $\{e\}$ is not a solution of $G(e)$. If $\{e\}(b) \uparrow$ or $\{e\}(b) \downarrow \notin \{a, c\}$, then $\{e\}$ is clearly not a solution. If $\{e\}(b) \downarrow$, then case 2 or 3 will occur, at which point $\{e\}$ will be forced not to be a solution of $G(e)$, by Lemma 5.12. $\square$

For recursive bipartite graphs the situation is even worse. Manaster and Rosenstein [119] have shown that there exist recursive bipartite graphs that satisfy Hall's condition but do not have any solution recursive in K . If we allow our bipartite graphs to have infinite degree, then the situation is far worse. Misercque [128] has shown that for every recursive tree T , there exists a recursive bipartite graph G , such that there is a degree preserving bijection between the infinite branches of T and the solutions of G . Since there exist recursive trees T where every infinite branch is not hyperarithmetical [142, p. 419, Corollary XLI(b)], there is a recursive bipartite graph where every solution is not hyperarithmetical.

5.3 How hard is it to determine if there is a recursive solution?

By Theorem 5.13, there are highly recursive bipartite graphs that satisfy Hall's condition, but have no recursive solution. We investigate how hard it is to determine if a particular highly recursive bipartite graph is of that type. By contrast the following promise problem is Π_1 -complete: (D, A) , where

$$D = \{e \mid e \text{ is the index of a highly recursive bipartite graph}\}$$

and

$$A = \{e \in D \mid \text{the graph represented by } e \text{ has a solution}\}.$$

Notation 5.14 Let HRB be the set of valid indices of highly recursive bipartite graphs.

Theorem 5.15 *The set*

$$\text{RECSOL} = \{e : e \in \text{HRB} \wedge G_e \text{ has a recursive solution}\}$$

is Σ_3 -complete.

Proof. For this proof, if $e = \langle e_1, e_2, e_3 \rangle$ is an index that determines a highly recursive bipartite graph, then we denote the graph that it determines by $G_e = (A_e, B_e, E_e)$. We abbreviate $(\exists t) \{e_1\}_t(x) = 1$ by $x \in A_e$, and adopt similar conventions for B_e and E_e .

RECSOL is the set of all triples $e = \langle e_1, e_2, e_3 \rangle$ of numbers in TOT such that there exists an i such that

- (i) $i \in \text{TOT}$, and
- (ii) $(\forall x, y) [(x \in A_e \wedge \{i\}(x) = y) \Rightarrow \{x, y\} \in E_e]$, and
- (iii) $(\forall x, z) [(x, z \in A_e \wedge x \neq z) \Rightarrow \{i\}(x) \neq \{i\}(z)]$

Clearly RECSOL is Σ_3 . To show RECSOL is Σ_3 -hard we show that $\text{COF} \leq_m \text{RECSOL}$. Given x , we 'try' to construct a highly recursive bipartite graph G that satisfies Hall's condition but does not have a recursive solution. We will always succeed in making G satisfy Hall's condition. If $x \in \text{COF}$, then our attempt will fail, in that G will have a recursive solution. If $x \notin \text{COF}$, then our attempt will succeed, in that G will not have a recursive solution.

We try to construct a highly recursive bipartite graph $G = (A, B, E)$ that satisfies the following requirements.

$R_e : \{e\} \text{ total} \Rightarrow \{e\} \text{ is not a solution of } G.$

Recursively partition $\mathbb{N}$ into infinite recursive sets $X_0, X_1, \dots$. We construct a highly recursive bipartite graph $G(e) = (A(e), B(e), E(e))$ such that the following hold.

- (a) $A(e) \cup B(e) = X_e,$
- (b) $G(e)$ has a solution,
- (c) If $\overline{W}_x \cap \{e, e+1, \dots\} \neq \emptyset$, then $\{e\}$ is not a solution of $G(e)$, and
- (d) If $\overline{W}_x \cap \{e, e+1, \dots\} = \emptyset$, then $G(e)$ has a recursive solution.

The union $G = \bigcup_{e \geq 0} G(e)$ is the desired graph. If there is an e_0 such that $\overline{W}_x \cap \{e_0, e_0 + 1, \dots\} = \emptyset$ then, for all $e \geq e_0$, $\overline{W}_x \cap \{e, e + 1, \dots\} = \emptyset$, hence $G(e)$ will have a recursive solution. From this we will be able to deduce that G has a recursive solution.

In our description of $G(e)$, whenever we need a vertex we take the least unused number of X_e . We denote the bipartite graph constructed by the end of stage e by $G(e, s) = (A(e, s), B(e, s), E(e, s))$. $G(e)$ is itself a union of disjoint line graphs. During each stage of the construction we are adding vertices to one of those line graphs, which we refer to as “the current component”.

Construction

STAGE 0:

Let $G(e, 0)$ be the line graph on (a, b, c) (the least three elements of X_e), interpreted as a bipartite graph by specifying $b \in A(e, 0)$. This will be the current component until it is explicitly changed. Set $\Gamma_{e,0} = e$.

STAGE $s + 1$:

If $\Gamma_{e,s} \notin W_{x,s}$, then set $\Gamma_{e,s+1} = \Gamma_{e,s}$ and proceed to work on R_e using the current component and current values of a, b and c , as in the construction in Theorem 5.13. If $\Gamma_{e,s} \in W_{x,s}$, then

- (i) Set $\Gamma_{e,s+1} = \mu z[z > \Gamma_{e,s} \wedge z \notin W_{x,s}]$.
- (ii) If the number of vertices in the current component is odd, then add a vertex to it (respecting whatever restraints there may be on which side you can add to).

- (iii) Let a, b, c be three new vertices (the least unused numbers in X_e), and start a new current component with the line graph on (a, b, c) , interpreted as a bipartite graph, by taking $b \in A(e, s+1)$.

End of the construction.

Let $G = \bigcup_{e \geq 0} G(e)$. For each e , the graph $G(e)$ is highly recursive by reasoning similar to that used in the proof of Theorem 5.13. It has a solution, since it consists of some number (possibly infinite) of finite graphs with an even number of vertices, and at most one infinite graph which is either the infinite two-way line graph or the infinite one-way line graph. Since all the algorithms for all the $G(e)$ are uniform, G is a highly recursive graph with a solution.

If $x \notin \text{COF}$ then, for all e , $\overline{W}_x \cap \{e, e+1, \dots\} \neq \emptyset$. Let y_e be the least element of $\overline{W}_x \cap \{e, e+1, \dots\}$. Then $\lim_{s \rightarrow \infty} \Gamma_{e,s} = y_e$, and in particular it exists. Eventually the attempt to make sure $\{e\}$ is not a solution of $G(e)$ will be acting on one component. In this case such efforts will succeed, as in the proof of Theorem 5.13. Now consider G . This graph has no recursive solution since, for all e , $\{e\}$ is not a solution of $G(e)$.

If $x \in \text{COF}$, then we show that G has a recursive solution. For almost all e , $\overline{W}_x \cap \{e, e+1, \dots\} = \emptyset$. Hence, for almost all e , $\lim_{s \rightarrow \infty} \Gamma_{e,s} = \infty$. Hence, for almost all e , all the components of $G(e)$ are finite and have an even number of vertices. Let S be the finite set of e such that $G(e)$ has an infinite component. For every $e \in S$, $G(e)$ has a finite number of finite components which have an even number of vertices, and one component that is either the two-way or one-way infinite line graph. Hence $G(e)$ has a recursive solution; let M_e be a machine that computes that solution. The set S and the indices of the machines M_e for all $e \in S$ are all finite information that can be incorporated into the following algorithm for a recursive solution of G . Given a number v , first find out if $v \in \bigcup_{e \geq 0} A(e)$. If $v \notin \bigcup_{e \geq 0} A(e)$, then we need not match v , so output 0 and stop. If $v \in \bigcup_{e \geq 0} A(e)$, then we find out if $v \in \bigcup_{e \in S} A(e)$, then find e such that $v \in A(e)$, and then output $M_e(v)$. If $v \notin \bigcup_{e \in S} A(e)$, then run the construction until all the vertices in the same component as v are in the graph (this will happen since every component of $G(e)$ is finite). Since the component is a line graph with an even number of vertices, there is a unique solution on it. Output the vertex to which v is matched. $\square$

5.4 Recursion-theoretic modification

Manaster and Rosenstein [119] showed that highly recursive bipartite graphs have solutions of low Turing degree.

Theorem 5.16 *If $G = (A, B, E)$ is a highly recursive bipartite graph that satisfies Hall's condition, then G has a solution M of low Turing degree.*

Proof. Let $A = \{a_1 < a_2 < \dots\}$. Consider the following recursive tree: The vertex $\sigma = (b_1, \dots, b_n)$ is on T iff

- (i) for all i ($1 \leq i \leq n$) we have $b_i \in nb_G(\{a_i\})$, and
- (ii) all the b_i 's are distinct (so the vertex σ represents a solution of the first n vertices of A into B , namely a_i maps to b_i).

We have

- (1) T is recursive,
- (2) T is recursively bounded by the function

$$f(n) = \max_{1 \leq i \leq n} nb_G(\{a_i\}).$$

- (3) any infinite branch of T is a solution of G ,
- (4) every solution of G is represented by some infinite branch of T , and
- (5) the set of infinite branches of T is nonempty (by the classical Hall's theorem and the previous item).

Since the branches of T form a nonempty Π_1^0 class, by Theorem 2.12 there exists an infinite low branch. This branch represents a solution of low degree. $\square$

Theorem 5.17 *If $G = (A, B, E)$ is a recursive bipartite graph that satisfies Hall's condition, then G has a solution M such that $M' \leq_T \emptyset''$.*

Proof. If $G = (A, B, E)$, then the function nb is recursive in K . Define a tree T as in the previous theorem. Note that this tree is recursive in K and is K -recursively bounded. By Theorem 2.13, there exists an infinite branch B such that $B' \leq_T \emptyset''$. This branch represents the desired solution. $\square$

5.5 Recursion-combinatorial modification

We now consider an effective version of Hall's theorem which is true. The modification is both recursion-theoretic and combinatorial.

Recall that by Theorem 5.5 a bipartite graph $G = (A, B, E)$ has a solution iff for all finite $X \subseteq A$,

$$|nb_G(X)| - |X| \geq 0.$$

A stronger condition would be to demand that $|nb_G(X)| - |X|$ is large for large $|X|$. In particular, if you want $|nb_G(X)| - |X| \geq n$ then there should be some n' such that $|X| \geq n'$ guarantees this. We formalize this:

Definition 5.18 A bipartite graph $G = (A, B, E)$ satisfies the *extended Hall's Condition* (e.H.c.) if there exists a function h such that $h(0) = 0$ and, for every finite $X \subseteq A$,

$$|X| \geq h(n) \Rightarrow |nb_G(X)| - |X| \geq n.$$

That is, to guarantee an 'expansion' of size n , take $|X| \geq h(n)$. Since $h(0) = 0$, e.H.c. implies Hall's condition.

Kierstead [96] proved the following effective version of Hall's theorem.

Theorem 5.19 *If $G = (A, B, E)$ is a highly recursive bipartite graph that satisfies e.H.c. with a recursive h , then G has a recursive solution. Moreover, given indices for G and h , one can effectively produce an index for a recursive solution.*

Proof. Let a_0 be the first element of A . We plan to match a_0 with some b_0 , define a function h' , show that $G' = (A - \{a_0\}, B - \{b_0\}, E')$, where

$$E' = \{\{x, y\} : \{x, y\} \in E, x \in A - \{a_0\}, y \in B - \{b_0\}\},$$

together with h' , satisfies the hypothesis of the theorem, and iterate. This will easily produce a recursive solution. Let

$$A_0 = \{x \in A : \text{there is a path from } x \text{ to } a_0 \text{ of length } \leq 2h(1)\},$$

$$B_0 = nb_G(A_0),$$

$$E_0 = \{\{x, y\} \in E : x \in A_0, y \in B_0\}.$$

Note that the vertices in B are of distance at most $2h(1) + 1$ from a_0 .

Let G_0 be the finite bipartite graph (A_0, B_0, E_0) . Clearly G_0 satisfies Hall's condition, so it has a solution. Let b_0 be the vertex to which a_0 is matched. Let h' be defined by

- (1) $h'(0) = 0$, and
- (2) $(\forall n \geq 1) h'(n) = h(n+1)$.

Let G' be as indicated above. We show that G' satisfies e.H.c. via h' .

Let $n \in \mathbb{N}$, $X \subseteq A - \{a_0\}$, X finite, and $|X| \geq h'(n)$. We show that $|nb_{G'}(X)| - |X| \geq n$. We need only consider X such that $(X, nb_{G'}(X), E'')$ (where $E'' \subseteq E'$ is the set of edges between X and $nb_{G'}(X)$) is connected. There are several cases, depending on n and X .

Case 1. $n \geq 1$. Then $|X| \geq h'(n) = h(n+1)$. Hence $|nb_G(X)| - |X| \geq n+1$. Hence $|nb_{G'}(X)| - |X| \geq n$.

Case 2. $n = 0$ and $b_0 \notin nb_G(X)$. Then $nb_{G'}(X) = nb_G(X)$. Hence $|nb_{G'}(X)| - |X| = |nb_G(X)| - |X| \geq 0$.

Case 3. $n = 0$ and $X \subseteq A_0$. Since (A_0, B_0, E_0) has a solution where a_0 maps to b_0 , $|nb_{G'}(X)| - |X| \geq 0$.

Case 4. $n = 0$, $b_0 \in nb_G(X)$, and there exists $a \in X - A_0$ (this is negation of Cases 1, 2, and 3). Since $b_0 \in nb_G(X)$, there exists a vertex $a' \in X$ such that $\{a', b_0\} \in E$. Since $(X, nb_{G'}(X), E'')$ is connected, there exists a path $\langle a = x_0, x_1, \dots, x_{2k} = a' \rangle$ in G' . Since G' is bipartite, $x_i \in X$ iff i is even, hence there are at least k elements of X . Since $\{a', b_0\} \in E$ and $\{a_0, b_0\} \in E$, there is a path of length $2k+1$ from a to a_0 . The shortest path from a to a_0 is of length $\geq 2h(1)+1$, hence $2k+1 \geq 2h(1)+1$, so $k \geq h(1)$. Thus $|X| \geq h(1)$. $\square$

Kierstead formulated the recursive e.H.c. to prove the following corollary.

Corollary 5.20 *Let $G = (A, B, E)$ be a highly recursive bipartite graph. Assume there exists d such that*

- (1) *for all $x \in A$, $\deg(x) > d$, and*
- (2) *for all $x \in B$, $\deg(x) \leq d$.*

Then G has a recursive solution.

Proof. We show that G together with the function $h(n) = dn$ satisfies the hypothesis of Theorem 5.19. Let X be a finite subset of A such that $|X| \geq dn$. We claim $|nb_G(X)| - |X| \geq n$. Consider the induced bipartite graph $(X, nb_G(X), E')$, where

$$E' = E \cap \{\{a, b\} \mid a \in X \wedge b \in nb_G(X)\}.$$

The number of edges is

$$\sum_{x \in X} \deg(x) \geq (d+1)|X|.$$

But it is also

$$\sum_{y \in nb_G(X)} \deg(y) \leq d|nb_G(X)|.$$

Simple algebra yields $|nb_G(X)| - |X| \geq n$. □

Kierstead [96] has also shown that the assumption that h is recursive cannot be dropped.

5.6 Miscellaneous

We state several results about recursive solutions without proof.

Manaster and Rosenstein [119] investigated whether some other conditions on G yield recursive solutions. They showed that

- (1) if a highly recursive bipartite graph G has a finite number of solutions, then all those solutions are recursive, and
- (2) if a recursive bipartite graph G has a finite number of solutions, then all those solutions are recursive in K .

However other conditions do not help:

- (i) there are highly recursive bipartite graphs where every vertex has degree 2 (this implies Hall's condition) which have no recursive solutions (this was extended to degree k in [120]),
- ii) there are decidable bipartite graphs (similar to decidable graphs, see Section 4.7.8) that satisfy Hall's condition but do not have recursive solutions.

McAloon [118] showed how to control the Turing degrees of solutions in graphs. He showed that there exists a recursive bipartite graph which satisfies Hall's condition and such that K is recursive in every solution. Along these lines, Manaster and Rosenstein (reported in [119]) showed that for every n , $1 \leq n \leq \aleph_0$, there exists a recursive bipartite graph with exactly n different solutions, and the n solutions are of n different Turing degrees. Manaster and Rosenstein also showed that for any Turing degree $\mathbf{a} \leq_T \mathbf{0}'$ there exists a recursive bipartite graph that has a unique solution M , and M is of Turing degree $\mathbf{a}$. This yields a contrast to highly recursive graphs since any highly recursive bipartite graph with a unique solution has a recursive solution.

Manaster and Rosenstein [119] examined symmetric solutions in highly recursive bipartite graphs. A symmetric solution is a solution in $G = (A, B, E)$ which is a solution of B to A as well as A to B . The results are similar to those for solutions, and thus we do not state them.

Manaster and Rosenstein also showed that for any Turing degree $\mathbf{a}$, there exists a recursive bipartite graph that has a unique solution M , and M is of Turing degree $\mathbf{a}$.

Misercque [128] has refined the above theorems by showing the following:

- (1) given a (highly) recursive bipartite graph G , there exists a (recursively bounded) recursive tree T such that there is a bijection between the infinite paths through T and the solutions of G which preserves degree of unsolvability,
- (2) the analogue of (1) also holds for symmetric solutions,
- (3) for every (recursively bounded) recursive tree T , there exists a (symmetric highly) recursive bipartite graph G such that there is a bijection between the infinite paths of T and the (symmetric) solutions of G , and
- (4) the analogue of (3) for arbitrary solutions is false (this disproved a conjecture of Jockusch and Soare from [90]).

Hirst [81] has proven several theorems about the proof-theoretic strength of Hall's Theorem. Several results in recursive solution theory can be derived as corollaries of his work.

6 Dilworth's Theorem for partial orders

We consider the infinite version of Dilworth's Theorem on partial orders. We present the finite and infinite versions, which are both due to Dilworth [47, 48], show that a recursive analogue of Dilworth's Theorem is false, and show that there is a recursion-theoretic modification that is true. We then show that there is a combinatorial modification that is true, and show that there is a modification that is both recursion-theoretic and combinatorial which is true, and finally state some miscellaneous results.

6.1 Definitions and classical version

Definition 6.1 A partial order $\mathcal{P}$ is a set P (called the *base set*) together with a relation $\leq$ that is transitive, reflexive, and anti-symmetric. The relation $<$ is defined by $x < y$ iff $x \leq y$ and $x \neq y$. If either $x \leq y$ or $y \leq x$, then x and y are *comparable*. If two elements x, y are not comparable, we denote this by $x \mid y$. A *chain* of $\mathcal{P} = \langle P, \leq \rangle$ is a set $C \subseteq P$ such that every pair of elements in C is comparable. A w -chain is a chain of size w . An *antichain* of $\langle P, \leq \rangle$ is a set $C \subseteq P$ such that every pair of elements in C is incomparable. A w -antichain is an antichain of size w . The *height* of $\langle P, \leq \rangle$ is the size of the largest chain. The *width* of $\langle P, \leq \rangle$, denoted $w(P)$, is the size of the largest antichain. If $w \in \mathbb{N}$, then a w -cover of $\langle P, \leq \rangle$ is a set of w disjoint chains such that every element of P is in some chain. We formally represent a w -cover as a function c from P to $\{1, \dots, w\}$ such that if $c(x) = c(y)$ then x is comparable to y .

Dilworth's theorem states that if the largest antichain of a partial order is of size w , then it can be covered with w chains. The first published proof is by Dilworth [47, 48], is by induction on the width, and is somewhat complicated.² Other proofs have been given by Dantzig and Hoffman [45], Fulkerson [57], Gallai and Milgram [59], and Perles [135]. The most efficient algorithm to find a covering of a finite partial order $\langle P, \leq \rangle$ uses the computational equivalence of finding a maximum matching in a bipartite graph to finding a minimal covering (see [44] or [39, pp. 339–341]) and runs in time $O(|P|^{2.5})$.

²Erdős [51] claims that Galai and Milgram had a complete proof of this in 1947, and that Galai did not want this known in his lifetime since he was modest and did not want to seem like he was bickering about priority.

We present a simple proof of Dilworth's theorem due to Perles [135].

Theorem 6.2 (Finite Dilworth's Theorem) *If $\mathcal{P} = \langle P, \leq \rangle$ is a finite partial order of width w , then $\mathcal{P}$ has a w -cover.*

Proof. We prove this by induction on $|P|$ (the size of P) for all w simultaneously. If $|P| = 1$ then the theorem is clear. Assume the theorem holds for all partial orders of size $\leq n - 1$. Let $\mathcal{P} = \langle P, \leq \rangle$ be a partial order such that $|P| = n$ and $w(\mathcal{P}) = w$. Let

$$P_{\max} = \{x \in P : (\forall y \in P)[x \text{ comparable to } y \Rightarrow y \leq x]\},$$

$$P_{\min} = \{x \in P : (\forall y \in P)[x \text{ comparable to } y \Rightarrow y \geq x]\},$$

There are two cases.

Case 1. There exists a w -antichain $A = \{a_1, \dots, a_w\}$ such that $A \neq P_{\max}$ and $A \neq P_{\min}$. Let $\mathcal{P}^+ = \langle P^+, \leq \rangle$ and $\mathcal{P}^- = \langle P^-, \leq \rangle$ where

$$P^+ = \{x \in P : (\exists i)[a_i \leq x]\},$$

$$P^- = \{x \in P : (\exists i)[a_i \geq x]\}$$

Clearly $P = P^+ \cup P^-$ (since $w(\mathcal{P}) = w$), $A = P^+ \cap P^-$ (since A is an antichain), $|P^+| < n$ (since $A \neq P_{\min}$), $|P^-| < n$ (since $A \neq P_{\max}$), and $w(\mathcal{P}^+) = w(\mathcal{P}^-) = w$. Apply the induction hypothesis to $\mathcal{P}^+$ and $\mathcal{P}^-$ to obtain w -coverings COV^+ of $\mathcal{P}^+$ and COV^- of $\mathcal{P}^-$. Assume, without loss of generality, that for all i , $1 \leq i \leq w$, $\text{COV}^+(a_i) = \text{COV}^-(a_i) = i$. Then $\text{COV}^+ \cup \text{COV}^-$ is a w -covering of $\mathcal{P}$.

Case 2. For all w -antichains A , either $A = P_{\max}$ or $A = P_{\min}$. Let C be a chain that has one endpoint in $P_{\max}$ and one in $P_{\min}$ (such easily exists — take an element of $P_{\max}$ and work your way down). Note that C intersects every w -antichain of $\mathcal{P}$ (i.e., intersects $P_{\max}$ and $P_{\min}$), hence the width of $\mathcal{P}' = \langle P - C, \leq \rangle$ is $\leq w - 1$. Since $|P - C| < n$ we can apply the induction hypothesis to $\mathcal{P}'$, to yield a $(w - 1)$ -covering of $\mathcal{P}'$. This covering, together with C , yields a w -covering of $\mathcal{P}$. $\square$

We now prove the infinite Dilworth's Theorem. We give a direct proof; it can also be proven by König's Lemma (Theorem 2.3).

Theorem 6.3 (Infinite Dilworth's Theorem) *If $\mathcal{P} = \langle P, \leq \rangle$ is a countable partial order of width w , then $\mathcal{P}$ has a w -cover.*

Proof. Assume, without loss of generality, that $P = \mathbb{N}$, though of course $\leq$ need not have any relation to the ordering of $\mathbb{N}$. Let $\mathcal{P}_s = \langle \{0, 1, \dots, s\}, \leq \rangle$. Let c_s be a w -covering of $\mathcal{P}_s$ (such exists by Theorem 6.2). We use c_s to define c , a w -covering of $\mathcal{P}$. Let

$$\begin{aligned} c(0) &= \mu i \left[(\exists^\infty s) c_s(0) = i \right], \\ c(n+1) &= \mu i \left[((\exists^\infty s) c_s(n+1) = i) \wedge \left(\bigwedge_{j=0}^n c_s(j) = c(j) \right) \right] \end{aligned}$$

It is easy to see that c is a w -covering. □

The proof of Theorem 6.3 given above is noneffective. To see if the proof could have been made effective we will look at a potential analogue. In order to state this analogue we need some definitions.

Definition 6.4 A partial order $\langle P, \leq \rangle$ is *recursive* if both the set P and the relation $\leq$ are recursive.

We will represent recursive partial orders $\langle P, \leq \rangle$ by the Turing machines that determine their base set and relation. An index for a recursive partial order will be an ordered pair $\langle e_1, e_2 \rangle$, such that e_1 is an index for a Turing machine that decides P , and e_2 is an index for a Turing machine that decides $\leq$.

Definition 6.5 An index e is a *valid index* for a recursive partial order if $e = \langle e_1, e_2 \rangle$ and the following hold.

- (i) $e_1 \in \text{TOT01}$. Let P denote $\{x : \{e_1\}(x) = 1\}$.
- (ii) $e_2 \in \text{TOT01}$.
- (iii) The relation defined by $x \leq y$ iff $\{e_2\}(x, y) = 1$, when restricted to $P \times P$, is a partial order on P .

The *partial order represented by e* is $\langle P, \leq \rangle$. We denote this partial order $\mathcal{P}_e$. Note that if $\{e_2\}(x, y) = 0$ and $\{e_2\}(y, x) = 0$, then x and y are incomparable.

Definition 6.6 If $\mathcal{P}$ is a recursive partial order, then the *recursive width* of $\mathcal{P}$ is the least w such that $\mathcal{P}$ can be recursively covered with w recursive chains. (Theorems 6.2 and 6.3 justify this definition.) We denote the recursive width of $\mathcal{P}$ by $w^r(\mathcal{P})$.

Potential Analogue 6.7 There is a recursive algorithm $\mathcal{A}$ that performs the following. Given an index e for a recursive partial order $\mathcal{P} = \langle P, \leq^P \rangle$ of width w , $\mathcal{A}$ outputs an index for a recursive w -covering of $\mathcal{P}$. A consequence is that all recursive partial orderings $\mathcal{P}$ have $w(\mathcal{P}) = w^r(\mathcal{P})$.

Kierstead [94] showed that this Potential Analogue is false, but that a combinatorial modification is true. We have a recursion-theoretic modification which is true. Schmerl (reported in [94]) has a modification that is both recursion-theoretic and combinatorial which is true. In summary, the following are true:

- (1) For every $w \geq 2$, there exists a recursive partial order $\mathcal{P}$ such that

$$w(\mathcal{P}) = w \quad \text{but} \quad w^r(\mathcal{P}) = \binom{w+1}{2}$$

(proved by Szemerédi and Trotter, reported in [97]). For the case of $w = 2$ closer bounds are known: every recursive partial order of width 2 can be recursively 6-covered; however, there exists a recursive partial order of width 2 that cannot be recursively 4-covered.

- (2) There is a recursive algorithm $\mathcal{A}$ that performs the following. Given an index e for a recursive partial order $\mathcal{P} = \langle P, \leq^P \rangle$ of width w , $\mathcal{A}$ outputs an index for a recursive $(\frac{5^w - 1}{4})$ -covering of $\mathcal{P}$ [94].
- (3) Every recursive partial order of width w has a w -covering that is low.
- (4) If $\mathcal{P}$ is a recursive locally finite partial order (defined in Section 6.6) then $w(\mathcal{P}) = w^r(\mathcal{P})$ (proven by Schmerl, reported in [94]). The proof does not yield an effective procedure to pass from indices of partial orders to indices of coverings.

We will prove subcases of (1) and (3) to give the reader the ideas involved. The full proofs use the same recursion-theoretic ideas, but are more complicated combinatorially. Items (2) and (4) will be proven completely.

We will need the following definitions.

Definition 6.8 A partial order $\mathcal{Q} = \langle Q, \leq^{\mathcal{Q}} \rangle$ *extends* a partial order $\mathcal{P} = \langle P, \leq^{\mathcal{P}} \rangle$ if $P \subseteq Q$ and, for all $x, y \in P$, if $x \leq^{\mathcal{P}} y$ then $x \leq^{\mathcal{Q}} y$. (Note that elements that are incomparable in $\langle P, \leq^{\mathcal{P}} \rangle$ might be comparable in $\langle Q, \leq^{\mathcal{Q}} \rangle$.)

Definition 6.9 Let $\mathcal{P}_0, \mathcal{P}_1, \dots$ be a (possibly finite) sequence of partial orders such that, for all i , $\mathcal{P}_{i+1}$ extends $\mathcal{P}_i$. Let $\mathcal{P}_j = \langle P_j, \leq_j \rangle$. Then *the union partial order of $\mathcal{P}_0, \mathcal{P}_1, \dots$ is*

$$\langle \bigcup_{j=0}^{\infty} P_j, \leq \rangle$$

where

$$x \leq y \quad \text{iff} \quad (\exists j) [x, y \in P_j \wedge x \leq_j y].$$

We denote this partial order by $\bigcup_j \mathcal{P}_j$.

6.2 Recursive Analogue is False

We show that there exists a recursive partial order of width w and recursive width $\binom{w+1}{2}$. Actually we prove something more general: for every u such that $w \leq u \leq \binom{w+1}{2}$ there is a recursive partial order $\mathcal{P}$ such that $w(\mathcal{P}) = w$ and $w^r(\mathcal{P}) = u$. The proof requires two lemmas; the first one is used in the second, and the second is similar in spirit to Lemma 4.13. The proof of the main theorem is similar in spirit to the proof of Theorem 4.15. The lemmas are more general than is needed for this section, but will be used again in Section 6.3.

Definition 6.10 Let $\{e\}$ be a Turing machine and let W be a set. If

$$(\forall x \in W) [\{e\}(x) \downarrow]$$

then $\{e\}(W)$ is defined to be $\{\{e\}(x) : x \in W\}$.

In the following lemma we show that, given a width $w \geq 1$ and a Turing machine $\{e\}$, we can construct a recursive partial order $\mathcal{P} = \langle P, \leq \rangle$ such that $w(\mathcal{P}) \leq w$, $w^r(\mathcal{P}) = w$, but $\{e\}$ will have a hard time covering $\mathcal{P}$. Formally, either

- (1) there is an $x \in P$ such that $\{e\}(x) \uparrow$,
- (2) there are $x, y \in P$ that are incomparable and $\{e\}(x) = \{e\}(y)$, or
- (3) there is a chain $A = \{a_w < a_{w-1} < \cdots < a_1\}$ such that $|\{e\}(A)| = w$.

If (1) or (2) occurs then $\{e\}$ is not a covering. If (3) happens then $\{e\}$ may still cover $\mathcal{P}$ but it has foolishly covered a single chain with w different chains.

Lemma 6.11 *Let $w \geq 1$, $\{e\}$ be a Turing machine, and X be an infinite recursive set. There exists a finite sequence of finite partial orders $\mathcal{P}_1, \dots, \mathcal{P}_r$ such that $r \leq w$, and the following hold. For notation $\mathcal{P}_j = \langle P_j, \leq_j \rangle$.*

- (i) *For every j , $1 \leq j \leq r$, $\mathcal{P}_j$ is a tree with one branch of length j , denoted $A_j = \{a_j < \cdots < a_1\}$, and leaves consisting of $B_j \cup \{a_j\}$, where B_j is an anti-chain, $B_j = \{b_1, \dots, b_k\}$ ($k \leq j-1$), and $A_j \cap B_j = \emptyset$. Every element of B_j is placed directly below some element of A_j , but no element of B_j is above any element of A_j . Since any antichain contains at most one element from A_j , $w(\mathcal{P}_j) \leq k+1 \leq j \leq w$.*

- (ii) *For every j , $1 \leq j \leq r-1$,*

- (1) *for all $x \in P_j$, $\{e\}(x) \downarrow$,*
- (2) *$\mathcal{P}_{j+1}$ can be obtained recursively from $\mathcal{P}_j$ (and the values of $\{e\}(x)$ for every $x \in P_j$),*
- (3) *$|\{e\}(A_j)| = |A_j| = j$, and $|\{e\}(B_j)| = |B_j|$.*

Also, $|\{e\}(B_r)| = |B_r|$.

- (iii) *For every j , $1 \leq j \leq r$,*

- (1) *$P_j \subseteq X$, and*
- (2) *canonical indices for the finite sets P_j and $\leq_j$ can be effectively obtained from e , j , w , and an index for X .*

Note that r is not needed.

- (iv) *For every j , $2 \leq j \leq r$,*

- (1) *$\mathcal{P}_j$ is an extension of $\mathcal{P}_{j-1}$,*
- (2) *$A_{j-1} \subseteq A_j$,*

- (3) $B_{j-1} \subseteq B_j$,
- (4) $a_j \in A_j - A_{j-1}$,
- (5) the only elements in $P_j - P_{j-1}$ are below and adjacent to a_{j-1} ,
and
- (6) $\{e\}(B_j) \subseteq \{e\}(A_{j-1})$.

(v) At least one of the following occurs.

- (1) $\{e\}(a_r) \uparrow$, (so $\{e\}$ cannot recursively cover $\mathcal{P}_r$).
- (2) $(\exists x, y \in A_r \cup B_r)$ such that $x \mid y$ and $\{e\}(x) \downarrow = \{e\}(y) \downarrow$ (so $\{e\}$ cannot recursively cover $\mathcal{P}_r$).
- (3) $r = w$ and $\{e\}(a_w) \downarrow \notin \{e\}(A_{w-1})$. By (2), $|\{e\}(A_{w-1})| = w - 1$, hence $|\{e\}(A_w)| = w$ (so if $\{e\}$ is trying to cover a partial order that includes $\mathcal{P}_r$, then it has just foolishly covered a single chain with w different chains).

- (vi) $\mathcal{P}_r$ is a recursive partial order. Moreover, an index for both P_r and $\leq_r$ can be obtained from e , w , and an index for X (note that we do not need r). The algorithm for P_r is as follows: given x , wait until x steps in the construction have gone by; if x has not entered the partial order by that step, it never will. The algorithm for $\leq_r$ is as follows: given x and y , wait until $\max\{x, y\}$ steps in the construction have gone by; if x and y are both in the partial order, then $x \leq y$ iff $x \leq y$ at that stage. (When an element enters the partial order, its relation to all the elements numerically less than it that are already in the partial order is known.)
- (vii) The following algorithm produces a recursive covering of $\mathcal{P}_r$ that uses $|B_r| + 1 \leq w$ covers. Map a_1 to 1; whenever p enters the partial order, map p to the least number that is not being used to cover some element incomparable to p (note that all elements already in the partial order will already be covered). This algorithm will map b_i to i . We will refer to this algorithm for covering as the greedy algorithm. It is easy to see that an index for the greedy algorithm can be effectively obtained from e , w , and an index for X . (What needs to be proved is that the greedy algorithm yields a w -covering.)

Proof. The Turing machine $\{e\}$ is fixed throughout this proof.

We prove this lemma by induction on w . Assume $w = 1$ and x is the least element of X . Let $\mathcal{P}_1 = \mathcal{P}_r = (\{x\}, \emptyset)$. If $\{e\}(x) \uparrow$, then $v.a$ is satisfied. If $\{e\}(x) \downarrow$, then $v.c$ is satisfied (vacuously). In either case conditions (i)–(vii) are easily seen to be satisfied. Note that $a_1 = x$.

Assume the lemma is true for w . We show it is true for $w + 1$. Let $\mathcal{P}_1, \dots, \mathcal{P}_r$ be the sequence of partial orders that exists via the induction hypothesis with parameter w .

If (v)(1) (or (v)(2)) holds for $\mathcal{P}_r$ with parameter w , then (v)(1) (or (v)(2)) holds for $\mathcal{P}_r$ with parameter $w + 1$. Hence the sequence $\mathcal{P}_1, \dots, \mathcal{P}_r$ is easily seen to satisfy (i)–(vii).

If (v)(3) holds for $\mathcal{P}_r$, then note that $r = w$ and let $A_w = \{a_1, \dots, a_w\}$ and $B_w = \{b_1, \dots, b_k\}$ be as in condition (1). Note that $\{e\}$ converges on all elements of A_w , and $|\{e\}(A_w)| = w$. We construct an extension of $\mathcal{P}_r$. Initialize as follows.

- (a) Set p to be a new number chosen to be the least element of X that is bigger than both any element used so far, and the number of steps spent on this construction so far (this is done to make the partial order recursive).
- (b) Set Z to be B_w . Place p under a_w and incomparable to all elements in Z (we do not yet say if this new element is in A_{w+1} or B_{w+1}).
- (c) Set k' to be k .

Be aware that p , Z and k' may change in the course of the construction. Note that all the elements in Z are pairwise incomparable. This will be true throughout the construction and easily provable inductively. Run $\{e\}(p)$. There are several cases; in all cases the only elements in $A_{w+1} - A_w$ or $B_{w+1} - B_w$ are those which we place there as follows.

- (1) $\{e\}(p) \not\downarrow \{e\}(A_w)$. Set a_{w+1} to p , A_{w+1} to $A_w \cup \{a_{w+1}\}$, and B_{w+1} to Z . Since $|\{e\}(A_{w+1})| = |\{e\}(A_w)| + 1 = w + 1$, condition (v)(3) is satisfied.
- (2) $\{e\}(p) \downarrow \in \{e\}(Z) \cap \{e\}(A_w)$. Set a_{w+1} to p , A_{w+1} to $A_w \cup \{a_{w+1}\}$, and B_{w+1} to Z . Since p is incomparable to all elements in Z , condition (v)(2) is satisfied. (Setting a_{w+1} to p is only a technicality to make condition (i) true.)

- (3) $\{e\}(p) \downarrow \in \{e\}(A_w) - \{e\}(Z)$. Set k' to $k' + 1$. Let $b_{k'}$ be p , and place p into Z . Designate a new number to be p , chosen to be the least element of X that is bigger than both any element used so far, and the number of steps spent on this construction so far (this is done to make the partial order recursive). Place p under a_w and incomparable to all elements of Z . Run $\{e\}(p)$ considering these cases (1), (2), (3), and (4). Note that every time case (3) occurs, $|Z|$ grows by one, and $|\{e\}(Z)| = |Z|$. Since $|A_w| = w$ (inductively using condition (i)), case (3) can occur at most $w - k$ times.
- (4) $\{e\}(p) \uparrow$. We (nonconstructively) set a_{w+1} to p , A_{w+1} to $A_w \cup \{a_{w+1}\}$, and B_{w+1} to Z . Condition (v)(1) is satisfied. (The $w + 2$ case is unaffected by this nonconstructiveness since $\{e\}(p)$ diverging yields a trivial induction step.)

In any case it is obvious how to define $\mathcal{P}_{r+1}$. It is easy to see that in any case conditions (i)–(vi) hold. We need to show that (vii) holds, namely that the greedy algorithm will $(w + 1)$ -cover $\mathcal{P}_{r+1}$. By the induction hypothesis the greedy algorithm $(k + 1)$ -covers $\mathcal{P}_r$, and covers b_i with i . Let $B_{w+1} = \{b_1, \dots, b_k, b_{k+1}, \dots, b_{k'}\}$. We prove, by induction on $i \geq k + 1$, that the greedy algorithm will cover b_i with i . Let $i \geq k + 1$. When b_i enters the partial order the greedy algorithm will cover it with i since $b_1, \dots, b_{i-1}$ are covered with $\{1, \dots, i-1\}$, and are the only elements that are incomparable to b_i . Note that there are $k' \leq w$ elements of B_{r+1} , and exactly one element in $A_{r+1} - A_r$ (namely a_{w+1}). The element a_{w+1} will be covered with $k' + 1 \leq w + 1$ when it enters. Hence the greedy algorithm provides a $(k' + 1)$ -covering with b_i getting covered with i . $\square$

In the following lemma, we show that given a width $w \geq 1$, a number u such that $w \leq u \leq \binom{w+1}{2}$, and a Turing machine $\{e\}$, we can construct a recursive partial order $\mathcal{Q} = \langle Q, \leq \rangle$ such that $w(\mathcal{Q}) \leq w$, $w^r(\mathcal{Q}) \leq u$, and $\{e\}$ is not a $(u - 1)$ -covering of $\mathcal{Q}$. Formally, either

- (1) there is an $x \in Q$ such that $\{e\}(x) \uparrow$,
- (2) there are $x, y \in Q$ that are incomparable and $\{e\}(x) = \{e\}(y)$, or
- (3) there is a set $W \subseteq Q$ such that $|\{e\}(W)| = u$.

If (1) or (2) occurs then $\{e\}$ is not a covering. If (3) happens, then $\{e\}$ may still cover $\mathcal{Q}$ but it has to use at least u different chains.

Lemma 6.12 *Let $w \geq 1$, $\{e\}$ be a Turing machine, and X be an infinite recursive set. Let u be any number such that $w \leq u \leq \binom{w+1}{2}$. There exists a finite sequence of finite partial orders $Q_1, \dots, Q_r$ such that the following hold. For notation, $Q_j = \langle Q_j, \leq_j \rangle$.*

(i) *For every j , $2 \leq j \leq r$, Q_j is an extension of Q_{j-1} . For each j , $1 \leq j \leq r$*

(1) $Q_j \subseteq X$, and

(2) *canonical indices for the finite sets Q_j and $\leq_j$ can be effectively computed given e, j, u, w , and an index for X .*

(ii) *For every j , $1 \leq j < r$,*

(1) *for all $x \in Q_j$, $\{e\}(x) \downarrow$,*

(2) Q_{j+1} *can be obtained recursively from Q_j (and the values of $\{e\}(x)$ for every $x \in Q_j$).*

(iii) *One of the following holds.*

(1) $(\exists x \in Q_r) [\{e\}(x) \uparrow]$ *(so $\{e\}$ cannot be a cover of Q_r).*

(2) $(\exists x, y \in Q_r) [\{e\}(x) \downarrow = \{e\}(y) \downarrow \text{ and } x \mid y]$ *(so $\{e\}$ cannot be a cover of Q_r).*

(3) $(\exists W \subseteq Q_r) [|W| = |\{e\}(W)| = u]$ *(so $\{e\}$ cannot be a $(u-1)$ -covering of Q_r).*

(iv) Q_r *is a recursive partial order. Moreover, an index for both Q_r and $\leq_r$ can be obtained from e, u, w , and an index for X (note that we do not need r). The algorithm is similar to that in Lemma 6.11 (vi).*

(v) $w(Q_r) \leq w$.

(vi) $w^r(Q_r) \leq u$. *Moreover, an index for a u -covering of Q_r can be effectively obtained from e, u, w , and an index for X .*

Proof. The Turing machine $\{e\}$ is fixed throughout this proof.

We prove this lemma by induction on w . If $w = 1$, then let $Q_1 = Q_r = (\{x\}, \emptyset)$ where x is the least element of X . If $\{e\}(x) \uparrow$, then (iii) (1) holds. If

$\{e\}(x) \downarrow$, then (iii) (3) holds with $W = \{x\}$. In either case conditions (i)–(vi) are easily seen to be satisfied.

Assume the lemma is true for w (and for all u with $w \leq u \leq \binom{w+1}{2}$). We show it for $w+1$ and any u such that $w+1 \leq u \leq \binom{w+2}{2}$. Recursively partition X into two infinite recursive sets Y and Z .

Let $\mathcal{P}_1, \dots, \mathcal{P}_{r_1}$ be the sequence that exists via Lemma 6.11 with parameters $w+1, Y$. For $1 \leq j \leq r_1$ let $\mathcal{Q}_j = \mathcal{P}_j$. If (v) (1) (or (v) (2)) of Lemma 6.11 holds for $\mathcal{P}_{r_1}$, then (iii) (1) (or (iii) (2)) holds for $\mathcal{Q}_{r_1}$. Since conditions (1)–(v) (of this lemma) are easily seen to be satisfied, we are almost done (we did not use the induction hypothesis). We will later see why (vi) is true.

We now assume that (v) (3) of Lemma 6.11 holds for $\mathcal{P}_{r_1}$. Let A denote A_{r_1} , B denote B_{r_1} , and $A = \{a_{w+1} < \dots < a_1\}$ ('<' is the order on P_{r_1}). Recall that for $1 \leq j \leq r_1$, we have set $\mathcal{Q}_j = \mathcal{P}_j$.

Note that $u \leq \binom{w+2}{2} = \binom{w+1}{2} + w + 1$. Let i be the least number such that $u - i \leq \binom{w+1}{2}$. Note that $0 \leq i \leq w + 1$. It is easy to see that $w \leq u - i \leq \binom{w+1}{2}$ (this uses $w + 1 \leq u$). Apply the induction hypothesis with parameters w , $u - i$, and Z to obtain the sequence $\mathcal{Q}'_1, \dots, \mathcal{Q}'_{r_2}$. If $i = 0$, then this sequence satisfies (i)–(v) and we are done. For the rest of the proof assume $i \geq 1$. We now construct $\mathcal{Q}_{r_1+1}, \dots, \mathcal{Q}_{r_1+r_2}$. For j , $r_1 + 1 \leq j \leq r_1 + r_2$, set $\mathcal{Q}_j = \langle \mathcal{Q}'_{j-r_1} \cup P_{r_1}, \leq_j \rangle$ where $\leq_j$ is defined as follows.

- (1) If $x, y \in \mathcal{Q}'_{j-r_1}$ (or P_{r_1}) then order as in $\mathcal{Q}'_{j-r_1}$ (or P_{r_1}).
- (2) If $x \in \{a_1, \dots, a_i\}$ and $y \in \mathcal{Q}'_{j-r_1}$, then set $x \mid y$.
- (3) If $x \in \{a_{i+1}, \dots, a_{w+1}\} \cup B$ and $y \in \mathcal{Q}'_{j-r_1}$, then $x <_j y$.

(Note that we needed $1 \leq i \leq w + 1$.) We examine the width of $\mathcal{Q}_{r_1+r_2}$. If C is an antichain of $\mathcal{Q}_{r_1+r_2}$, then one of the following occurs.

- (a) $C \cap B \neq \emptyset$, so $C \cap \mathcal{Q}'_{r_2} = \emptyset$. Hence $C \subseteq P_{r_1}$, and since P_{r_1} has width at most $w + 1$, $|C| \leq w + 1$,
- (b) $C \cap B = \emptyset$, so $C \cap P_{r_1} \subseteq A$. Since A is a chain, $|C \cap P_{r_1}| \leq 1$. Since $\mathcal{Q}'_{r_2}$ has width at most w , $|C \cap \mathcal{Q}'_{r_2}| \leq w$. Hence $|C| \leq w + 1$.

Since both cases lead to $|C| \leq w + 1$, $\mathcal{Q}_{r_1+r_2}$ has width $\leq w + 1$.

We now prove that condition (iii) holds for $\mathcal{Q}_{r_1+r_2}$. If (iii) (1) (or (iii) (2)) holds for $\mathcal{Q}'_{r_2}$ then (iii) (1) (or (iii) (2)) holds for $\mathcal{Q}_{r_1+r_2}$. If (iii) (3) holds for $\mathcal{Q}'_{r_2}$, then let $W' \subseteq \mathcal{Q}'_{r_2}$ be such that $|\{e\}(W')| = u - i$. If $\{e\}(W') \cap \{e\}(\{a_1, \dots, a_i\}) \neq \emptyset$, then there exist x, y such that $x|y$ and $\{e\}(x) = \{e\}(y)$, so (iii) (2) holds. If $\{e\}(W') \cap \{e\}(\{a_1, \dots, a_i\}) = \emptyset$, then set W to be $W' \cup \{a_1, \dots, a_i\}$. Note that $|\{e\}(W)| = |\{e\}(W')| + i = u - i + i = u$. Hence (iii) (3) holds.

By Lemma 6.11, we can effectively obtain, from e, w , and an index for Y , an index for a $(w+1)$ -covering COV_1 of $\mathcal{P}_{r_1}$. Inductively, we can effectively obtain, from $e, u-i, w$, and an index for Z , an index for a recursive $(u-i)$ -covering COV_2 of $\mathcal{Q}'_{r_2}$. (Since we can obtain i from u and w , we can also obtain the index for COV_2 from e, u, w , and an index for Z .) Recall that, by the definition of a cover, COV_2 has range $\{1, \dots, u-i\}$. We define a u -covering COV of $\mathcal{Q}_{r_1+r_2}$ via

$$\text{COV}(x) = \begin{cases} \text{COV}_1(x) & \text{if } x \in \mathcal{P}_{r_1}, \\ \text{COV}_2(x) + i & \text{if } x \in \mathcal{Q}'_{r_2}. \end{cases}$$

To compute $\text{COV}(x)$, do the following. Given x , first find if $x \in Y$ or $x \in Z$ (if it is in neither then stop and output 1). If it is in Y (or Z) then run the construction of the sequence of $\mathcal{P}_j$ (or sequence of $\mathcal{Q}'_j$) until either x appears, or the number of steps used is larger than x (in which case x never will appear, so output 1). If x does appear then compute and output $\text{COV}_1(x)$ (or $\text{COV}_2(x) + i$). Note that to construct an index for this function, we only needed indices for COV_1 and COV_2 , we did not need to know the manner in which the sequence of $\mathcal{P}_j$ or $\mathcal{Q}'_j$ succeeded in meeting its requirements. Hence we can effectively obtain this index even if the sequence of $\mathcal{P}$'s satisfies (v) (3) of Lemma 6.11.

It is clear that the range of COV is a subset of $\{1, \dots, u\}$. We show that COV is a covering. If $\text{COV}(x) = \text{COV}(y)$, then either

- (1) $x, y \in \mathcal{P}_{r_1}$ (or $x, y \in \mathcal{Q}'_{r_2}$), in which case x and y are comparable since $\text{COV}_1(x) = \text{COV}_1(y)$ (or $\text{COV}_2(x) = \text{COV}_2(y)$),
- (2) $x \in \mathcal{P}_{r_1} - \{a_1, \dots, a_i\}$ and $y \in \mathcal{Q}'_{r_2}$ (or vice versa), in which case $x <_{r_1+r_2} y$ by definition of $<_{r_1+r_2}$.

(The case $x \in \{a_1, \dots, a_i\}$ and $y \in \mathcal{Q}'_{r_2}$ cannot occur since then $\text{COV}(x) \in \{1, \dots, i\}$ and $\text{COV}(y) \in \{i+1, \dots, u\}$.) Hence COV is a u -cover. $\square$

The following lemma is similar to Lemma 6.12 except that we make $w(\mathcal{Q}) = w$ instead of $w(\mathcal{Q}) \leq w$.

Lemma 6.13 *Let $w \geq 1$, $\{e\}$ be a Turing machine, and X be an infinite recursive set. Let u be any number such that $w \leq u \leq \binom{w+1}{2}$. There exists a finite sequence of finite partial orders $\mathcal{Q}_1, \dots, \mathcal{Q}_r$ such that the following hold. For notation, $\mathcal{Q}_j = \langle Q_j, \leq_j \rangle$.*

- (i) *There exists a w -antichain A such that, for all j , $A \subseteq Q_j$, and all elements of A are less than all elements of $Q_j - A$. For each j , $1 \leq j \leq r$,*
 - (1) $Q_j \subseteq X$, and
 - (2) *canonical indices for the finite sets Q_j and $\leq_j$ can be effectively computed given e, i, j , and an index for X .*
- (ii) *For every j , $1 \leq j \leq r-1$, Q_{j+1} can be effectively obtained from Q_j and the values of $\{e\}(x)$ for every $x \in Q_j - A$.*
- (iii) $\{e\}$ *is not a $(u-1)$ -covering of $\mathcal{Q}_r$.*
- (iv) $w(\mathcal{Q}_r) = w$, *this is the difference between this Lemma and Lemma 6.12.*
- (v) $\mathcal{Q}_r$ *is a recursive partial order. Moreover, an index for both Q_r and $\leq_r$ can be obtained from e, u, w , and an index for X (note that we do not need r). The algorithm is similar to that in Lemma 6.11 (vi).*
- (vi) $w^r(\mathcal{Q}_r) \leq u$. *Moreover, an index for a u -covering of $\mathcal{Q}_r$ can be effectively obtained from e, u, w , and an index for X .*

Proof. Let $A = \{x_1, \dots, x_w\}$, the first w elements of X . Let $\mathcal{Q}'_1, \dots, \mathcal{Q}'_r$ be the sequence obtained by applying Lemma 6.12 to parameters e, u, w , and $X - \{x_1, \dots, x_w\}$. For notation, $\mathcal{Q}'_j = \langle Q'_j, \leq'_j \rangle$. For all j , $1 \leq j \leq r$, let $\mathcal{Q}_j = \langle Q'_j \cup A, \leq_j \rangle$ where $\leq_j$ is defined by the following:

- (1) if $x, y \in A$ then $x \mid y$,
- (2) if $x, y \in Q'_j$ then $x \leq_j y$ iff $x \leq'_j y$, and
- (3) if $x \in A$ and $y \in Q'_j$ then $x \leq_j y$.

□

In Lemma 6.13 we showed, given e , u , and w , how to create a recursive partial order $\mathcal{P}$ such that $w(\mathcal{P}) = w$, $w^r(\mathcal{P}) \leq u$, and $\{e\}$ does not $(u-1)$ -cover $\mathcal{P}$. We now combine all these partial orders to get a partial order that has width w , recursive width $\leq u$, and cannot be $(u-1)$ -covered by any $\{e\}$. Hence its recursive width is exactly u .

Theorem 6.14 *Let $w \geq 2$ and u be such that $w \leq u \leq \binom{w+1}{2}$. Let X be an infinite recursive set. There exists a recursive partial order $\mathcal{P} = \langle P, \leq^P \rangle$ such that $w(\mathcal{P}) = w$, $w^r(\mathcal{P}) = u$, and $P \subseteq X$. (Note that if $w \in \{0, 1\}$, then for all recursive partial orders $\mathcal{P}$ such that $w(\mathcal{P}) = w$, we have $w^r(\mathcal{P}) = w(\mathcal{P})$.)*

Proof. Let $X = \bigcup_{e \geq 0} X_e$ be a recursive partition of X into infinite sets. Let $\mathcal{Q}(e) = \langle Q_e, \leq_e \rangle$ be the partial order constructed in Lemma 6.13 using parameters e , u , w , and X_e . Let

$$\mathcal{P} = \langle \bigcup_{e \geq 0} Q(e), \leq \rangle,$$

where $\leq$ is defined by

- (1) if $(\exists e)[x, y \in Q(e)]$, then $x \leq y$ iff $x \leq_e y$,
- (2) if $x \in Q(e_1)$ and $y \in Q(e_2)$, then $x \leq y$ iff x is bigger than y numerically.

Clearly $\mathcal{Q}$ is a recursive partial order and $w(\mathcal{Q}) = w$. Since for all e , $\{e\}$ is not a $(u-1)$ -covering of $\mathcal{Q}(e)$, $w^r(\mathcal{Q}) \geq u$. Since for all e , $\mathcal{Q}(e)$ is recursively u -coverable in a uniform way, $w^r(\mathcal{Q}) \leq u$. Combining the two inequalities yields $w^r(\mathcal{Q}) = u$. $\square$

6.3 How hard is it to determine $w^r(\mathcal{P})$?

In this section we show that, even if $w(\mathcal{P})$ is known, and $w^r(\mathcal{P})$ is narrowed down to two prespecified values, it is Σ_3 -complete to determine $w^r(\mathcal{P})$. By contrast, the following promise problem is Π_1 -complete: (D, A) , where

$$D = \{e \mid e \text{ is the index of a recursive partial order}\}$$

and

$$A = \{e \in D \mid \text{the partial order represented by } e \text{ has width } \leq w\}.$$

The next lemma ‘slows down’ the construction of Lemma 6.12.

Lemma 6.15 *Let $w \geq 1$. Let $\{e\}$ be a Turing machine, and X an infinite recursive set. Let u be any number such that $w \leq u \leq \binom{w+1}{2}$. There exists an infinite sequence of (not necessarily distinct) partial orders $\mathcal{R}_1, \mathcal{R}_2, \dots$ such that the following hold. For notation, $\mathcal{R}_s = \langle R_s, \leq_s \rangle$.*

- (i) $\mathcal{R}_{s+1}$ is an extension of $\mathcal{R}_s$.
- (ii) For all s , $R_s \subseteq X$ and $w(\mathcal{R}_s) = w$.
- (iii) Given e, u, w , and an index for X , one can effectively find canonical indices for the finite sets R_s and $\leq_s$.
- (iv) There exists a finite partial order $\mathcal{R} = \langle R, \leq \rangle$ and a number t , such that $\mathcal{R} = \mathcal{R}_t$, and $(\forall s \geq t) \mathcal{R}_s = \mathcal{R}$. We call this partial order $\lim_{s \rightarrow \infty} \mathcal{R}_s$.
- (v) $\mathcal{R}$ is not $(u-1)$ -covered by $\{e\}$.
- (vi) $\mathcal{R}$ is a recursive partial order. Moreover, an index for both R and $\leq$ can be obtained from e, u, w , and an index for X . The algorithm is similar to that in Lemma 6.11 (vi).
- (vii) Let x, y, s be such that $x, y \in R_s$, and s is the least such number. Then for all $t \geq s$, $x \leq_s y$ iff $x \leq_t y$, i.e., if elements are initially incomparable then they remain incomparable.
- (viii) $w(\mathcal{R}) = w$.
- (ix) $w^r(\mathcal{R}) \leq u$. Moreover, given e, u, w and an index for X , one can effectively find an index for a recursive u -covering of $\mathcal{R}$.

Proof. Apply Lemma 6.13 to the parameters e, u, w , and X . Break the construction of $\mathcal{Q}_r$ into stages such that at each stage, either nothing is added to the partial order (e.g., one more step of the relevant Turing machine was run and did not converge), or an element is added and its relation with everything that is already in the partial order is established. Let $\mathcal{R}_s$ be the partial order produced at the end of stage s . It is easy to see that (i)–(ix) are satisfied. $\square$

The next lemma is a ‘parameterized version’ of Lemma 6.13. Given e, u, w , and a parameter y , we construct a partial order $\mathcal{P}$ such that

- (1) $w(\mathcal{P}) = w$,
- (2) $w^r(\mathcal{P}) \leq u$, and
- (3) if $y \notin \text{TOT}$, then $\mathcal{P}$ is not $(u-1)$ -covered by $\{e\}$, and if $y \in \text{TOT}$, then $\mathcal{P}$ will be recursively w -covered (in this case we do not care about what $\{e\}$ does).

Lemma 6.16 *Let $w \geq 1$. Let $\{e\}$ be a Turing machine, and X an infinite recursive set. Let u be any number such that $w \leq u \leq \binom{w+1}{2}$. Let $y \in \mathbb{N}$. There exists a recursive partial order $\mathcal{P} = (P, \leq)$, which depends on y , such that the following hold.*

- (i) $P \subseteq X$.
- (ii) Given e, u, w, y , and an index for X , one can effectively find an index for $\mathcal{P}$. The algorithm is similar to the one used in Lemma 6.11 (vi).
- (iii) $\mathcal{P}$ consists of a (possibly finite) set of finite partial orders

$$\mathcal{P}^1 = \langle P^1, \leq^1 \rangle, \mathcal{P}^2 = \langle P^2, \leq^2 \rangle, \mathcal{P}^3 = \langle P^3, \leq^3 \rangle, \dots,$$

such that for all $i \neq j$, $P^i \cap P^j = \emptyset$, and all elements of P^j are less than (using the order $\leq$ of $\mathcal{P}$) all elements of P^{j+1} . The $\mathcal{P}^j$'s are called constituents. The function that takes $\langle j, x \rangle$ and tells whether $x \in P^j$, is recursive.

- (iv) If $y \notin \text{TOT}$, then
 - (a) $\mathcal{P}$ consists of a finite number of constituents, and
 - (b) $\mathcal{P}$ is not $(u-1)$ -covered by $\{e\}$.
- (v) If $y \in \text{TOT}$, then
 - (a) $\mathcal{P}$ consists of an infinite number of constituents, and
 - (b) given e, u, w, y , and an index for X , and $p \in P$, one can effectively find the constituent containing p (i.e., find all the elements in the constituent and how they relate to p).
 - (c) $w^r(\mathcal{P}) = w$. This will follow from (a), (b), the finiteness of the constituents, and $w(\mathcal{P}) = w$ (the next item).

(vi) $w(\mathcal{P}) = w$.

(vii) $w^r(\mathcal{P}) \leq u$. Moreover, given e, u, w, y , and an index for X , one can effectively find an index for a recursive u -covering of $\mathcal{P}$.

Proof. We consider e, u, w, y , and X fixed throughout this proof. Let $X = \bigcup_{j=0}^{\infty} X_j$ be a recursive partition of X into an infinite number of infinite recursive sets. Let $\mathcal{R}_1(j), \mathcal{R}_2(j), \dots$ be the sequence of partial orders obtained by applying Lemma 6.15 to parameters e, u, w and X_j . For notation, $\mathcal{R}_s(j) = \langle R_s(j), \leq_{s,j} \rangle$. We use these partial orders to construct $\mathcal{P}$ in stages. We denote the partial order at the end of stage s by $\mathcal{P}_s = \langle P_s, \leq_s \rangle$.

Construction

STAGE 0:

$$\mathcal{P}_0 = \mathcal{R}_1(0), j_0 = 0, \text{ and } k_0 = 0.$$

STAGE $s + 1$:

Assume inductively that $\mathcal{P}_s = \langle \bigcup_{j=0}^{j_s} P_s^j, \leq_s \rangle$, $\mathcal{P}_s^{j_s} = \mathcal{R}_s(j_s)$, and that for all j , $0 \leq j \leq j_s - 1$, all the elements of $\mathcal{P}_s^j$ are $\leq_s$ -less than all the elements of $\mathcal{P}_s^{j+1}$. Let k_{s+1} be the least element that is not in $W_{y,s}$. If $k_s = k_{s+1}$ then set $j_{s+1} = j_s$, else set $j_{s+1} = j_s + 1$. In either case set

$$(1) \text{ for all } j < j_{s+1}, \mathcal{P}_{s+1}^j = \mathcal{P}_s^j, \text{ and}$$

$$(2) \mathcal{P}_{s+1}^{j_{s+1}} = \mathcal{R}_{s+1}(j_{s+1}).$$

(We refer to $\mathcal{R}_{s+1}(j_{s+1})$ as the *current partial suborder*.) Define $\leq_{s+1}$ as follows.

- (i) $x_1, x_2 \in \bigcup_{j=0}^{j_s} P_s^j$. So x_1, x_2 have been placed into the partial order in a previous stage. Set $x_1 \leq_{s+1} x_2$ iff $x_1 \leq_s x_2$.
- (ii) $x_1, x_2 \in R_{s+1}(j_{s+1})$. Set $x_1 \leq_{s+1} x_2$ iff $x_1 \leq_{s+1, j_{s+1}} x_2$. (If $x_1, x_2 \in \bigcup_{j=0}^{j_s} P_s^j$ then this is not in conflict with case (i). The relationship between x_1 and x_2 would have been set at a previous stage $s' \leq s$ via $x_1 \leq_{s'} x_2$ iff $x_1 \leq_{s', j_{s'}} x_2$ where $j_{s+1} = j_{s'}$. Note that by (vii) of Lemma 6.15, the relationship between x_1 and x_2 cannot change.)
- (iii) $x_1 \notin R_{s+1}(j_{s+1})$ and $x_2 \in R_{s+1}(j_{s+1})$. So x_1 is not in the current partial suborder, but x_2 is. Set $x_1 \leq_{s+1} x_2$. (If $x_1, x_2 \in \bigcup_{j=0}^{j_s} P_s^j$ then this is not in conflict with (i) since x_1 would have been set less than x_2 when x_2 enters the partial order, and via case (iii).)

Note that if $k_{s+1} = k_s$ then we do not add any more constituents, we just add to the most recent one; and if $k_s \neq k_{s+1}$ then we create a new constituent and will never add to the previous constituents.

Set

$$\mathcal{P}_{s+1} = \langle \bigcup_{j=0}^{j_{s+1}} P_{s+1}^j, \leq_{s+1} \rangle.$$

End of the construction.

Let $\mathcal{P} = \bigcup_s \mathcal{P}_s$. It is clear that $\mathcal{P}$ satisfies (i) and (ii). Since for every j both $\mathcal{R}_s(j)$ and $\mathcal{R} = \lim_{s \rightarrow \infty} \mathcal{R}_s(j)$ are finite, (iii) holds. By Lemma 6.15 each constituent of $\mathcal{P}$ is w -coverable therefore $\mathcal{P}$ has width w . Hence (vi) holds.

Assume $y \notin \text{TOT}$. Let k be the least element of $\overline{W}_y$. Let t be the least stage such that $0, 1, \dots, k-1 \in W_{y,t}$. For all $s > t$, $j_s = j_{t+1}$; therefore $\mathcal{P}$ consists of a finite number of finite partial orders of the form $\mathcal{R}_{s'}(j')$ (where $s' < t$ and $j' < j_{t+1}$) along with $\mathcal{R} = \lim_{s \rightarrow \infty} \mathcal{R}_s(j_{t+1})$. Hence (iv) (a) holds. By Lemma 6.15, $\mathcal{R}$ is not $(u-1)$ -covered by $\{e\}$, hence (iv) (b) holds.

Assume $y \in \text{TOT}$. Since $W_y = \mathbb{N}$, $\lim_{s \rightarrow \infty} j_s = \infty$. During every stage s such that $j_s \neq j_{s+1}$ a new constituent is created; therefore $\mathcal{P}$ consists of an infinite number of constituents. Hence (v) (a) holds.

To establish (v) (b) we show, given $p \in P$, how to find all the elements in the constituent containing p . Run the construction until $j, s \in \mathbb{N}$ are found such that p is an element of $\mathcal{R}_s(j)$ (this will happen since $p \in P$). Run the construction further until t is found such that $j < j_t$ (this will happen since $y \in \text{TOT}$). The constituent of $\mathcal{P}_t$ that contains p is the constituent of $\mathcal{P}$ that contains p .

To establish (v) (c), we show that $w^r(\mathcal{P}) = w$. Given a number p , first test if $p \in P$. If $p \notin P$ then output 1 and halt (we need not cover it). If $p \in P$ then, using (v) (b), find all the elements of the constituent containing p . By (vi) this constituent has width w . Let c be the least lexicographical w -covering of this constituent. Output $c(p)$.

To establish (vii), we have to effectively find an index for a u -covering of $\mathcal{P}$ from e, u, w, y , and an index for X . Since $X = \bigcup_{j=0}^{\infty}$ is a recursive partition, we need only find, for each j , an index for the construction restricted to X_j ,

which we denote $\mathcal{P}[j]$. Let $\mathcal{R}[j]$ be the recursive partial order obtained by applying Lemma 6.15 with parameters e , u , w , and X_j . Note that

- (1) $\mathcal{P}[j]$ is a suborder of $\mathcal{R}[j]$, and
- (2) we can effectively find an index e_j for a u -covering of $\mathcal{R}[j]$ from e , u , w , and an index for X_j .

The index e_j restricted to the subset of X_j that is actually used, is an index for a u -covering of $\mathcal{P}[j]$. Note that this index is obtained without knowing if $y \in \text{TOT}$. $\square$

Theorem 6.17 *Let $w \geq 2$. Let u be such that $w < u \leq \binom{w+1}{2}$. Let D be the set of all indices of recursive partial orders $\mathcal{P}$ such that $w(\mathcal{P}) = w$ and $w^r(\mathcal{P}) \in \{u, w\}$. Let $\text{RWIDTH}_{u,w}$ be the 0-1 valued partial function defined by*

$$\text{RWIDTH}_{u,w}(e) = \begin{cases} 1 & \text{if } e \in D \text{ and } w^r(\mathcal{P}_e) = w, \\ 0 & \text{if } e \in D \text{ and } w^r(\mathcal{P}_e) = u, \\ \text{undefined} & \text{if } e \notin D. \end{cases}$$

The promise problem $(D, \text{RWIDTH}_{u,w})$ is Σ_3 -complete.

Proof. The following is a Σ_3 solution for $(D, \text{RWIDTH}_{u,w})$.

A_w is the set of ordered pairs $\langle e_1, e_2 \rangle \in \text{TOT01}$ such that there exists an i such that

- (i) $i \in \text{TOT}_w$, and
- (ii) $(\forall x, y) [(\{i\}(x) = \{i\}(y) \wedge \{e_1\}(x) = \{e_1\}(y) = 1) \\ \Rightarrow (\{e_2\}(x, y) = 1 \vee \{e_2\}(y, x) = 1)]$

(Recall that x, y are comparable iff either $\{e_2\}(x, y) = 1$ or $\{e_2\}(y, x) = 1$.)

We show that $(D, \text{RWIDTH}_{u,w})$ is Σ_3 -hard by showing that if A is a solution to $(D, \text{RWIDTH}_{u,w})$, then $\text{COF} \leq_m A$. Given x , we construct a recursive partial order $\mathcal{P}(x) = \mathcal{P}$ such that $w(\mathcal{P}) = w$ and

$$x \in \text{COF} \Rightarrow w^r(\mathcal{P}) = w, \quad \text{and}$$

$$x \notin \text{COF} \Rightarrow w^r(\mathcal{P}) = u.$$

We use a modification of the construction in Theorem 6.14 of a recursive partial order which has width w but recursive width u . In this modification we weave the set W_x into the construction in such a way that if W_x is cofinite then the construction fails and $w^r(\mathcal{P}) = w$; and if W_x is not cofinite then the construction succeeds and $w^r(\mathcal{P}) = u$.

Let $\mathbb{N} = \bigcup_{e=0}^{\infty} X_e$ be a recursive partition of $\mathbb{N}$ into an infinite number of infinite recursive sets. Let y_e be defined such that

$$y_e \in \text{TOT} \quad \text{iff} \quad \{e, e+1, \dots\} \subseteq W_x$$

(it is easy to construct y_e from e). Let $\mathcal{P}(e) = \langle P(e), \leq_e \rangle$ be the recursive partial order obtained by applying Lemma 6.16 to e, u, w, X_e and y_e .

Let $\mathcal{P} = \langle \bigcup_{e=0}^{\infty} P(e), \leq \rangle$, where $\leq$ is defined as follows:

- (1) if $x_1, x_2 \in P(e)$, then $x_1 \leq x_2$ iff $x_1 \leq_e x_2$,
- (2) if $x_1 \in P(e_1)$ and $x_2 \in P(e_2)$, then $x_1 \leq x_2$ iff e_1 is numerically less than e_2 .

Clearly $\mathcal{P}$ is recursive and $w(\mathcal{P}) = w$.

If $x \notin \text{COF}$, then for all e we have $y_e \notin \text{TOT}$. Hence, by Lemma 6.16, for all e , $\mathcal{P}(e)$ is not $(u-1)$ -covered by $\{e\}$. Therefore $w^r(\mathcal{P}) \geq u$. By Lemma 6.16, the partial orders $\mathcal{P}(e)$ are recursively u -coverable, and an index for a recursive u -covering can be obtained from e, u, w, y_e , and an index for X_e . Hence $w^r(\mathcal{P}) \leq u$. Combining these two inequalities yields $w^r(\mathcal{P}) = u$.

If $x \in \text{COF}$, then $S = \{e \mid y_e \notin \text{TOT}\}$ is finite. Let $P' = \bigcup_{e \in S} P(e)$ and $P'' = \bigcup_{e \notin S} P(e)$. Let $\mathcal{P}' = \langle P', \leq' \rangle$ (or $\mathcal{P}'' = \langle P'', \leq'' \rangle$) where $\leq'$ (or $\leq''$) is the restriction of $\leq$ to P' (or P''). We show that $\mathcal{P}$ is recursively w -coverable by showing that $\mathcal{P}'$ and $\mathcal{P}''$ are recursively w -coverable (and using that $P = P' \cup P''$ is a recursive partition of P).

If $e \in S$, then $y_e \notin \text{TOT}$, and so by Lemma 6.16, $\mathcal{P}(e)$ is finite and $w(\mathcal{P}(e)) = w$. Since S is finite, $\mathcal{P}'$ is a finite w -coverable partial order. Hence $w^r(\mathcal{P}') = w$.

If $e \notin S$, then $y_e \in \text{TOT}$, so by Lemma 6.16, $w^r(\mathcal{P}'') = w$. □

6.4 Combinatorial modifications

Kierstead [94] proved that every recursive partial order of width w has recursive width $\leq \frac{5^w - 1}{4}$. In Section 6.4.1, we present the $w = 2$ case of this theorem in detail, i.e., we show that every recursive partial order of width 2 can be recursively 6-covered. We then make remarks about how the proof for general w goes. We do not claim that from this one could reconstruct the proof for general w . In Section 6.4.2 we examine a modification where less information about the partial order is given; we provide no proofs.

By Theorem 6.14 there exist a recursive partial order of width 2 that cannot be covered by 3 recursive chains. Kierstead has shown that there exists a recursive partial order of width 2 that cannot be covered by 4 recursive chains

Also note that the lower bound of $\binom{w+1}{2}$ given in Theorem 6.14 cannot be tight, since it fails for $w = 2$. The exact bound is unknown. It is open to find a w such that one can always recursively cover a partial order of width w with $< \frac{5^w - 1}{4}$ chains.

6.4.1 Bounding the recursive width

Notation 6.18 We often deal with several partial orders at the same time. In this case, each partial order we deal with will have a superscript on the ‘ $\leq$ ’ symbol. Hence we use $\langle P, \leq^P \rangle$ for a partial order, $\leq^{\mathbb{N}}$ for the numerical order on the natural numbers, and $\leq^*$ for an order that we define. To indicate which order we are using, we use terms like “ $\mathbb{N}$ -greater than” or “ $*$ -comparable”.

Theorem 6.19 *If $\mathcal{P} = \langle P, \leq^P \rangle$ is a recursive partial order of width 2, then $\mathcal{P}$ has recursive width ≤ 6 . Moreover, given an index for $\langle P, \leq^P \rangle$, one can recursively find an index for a recursive 6-covering of $\mathcal{P}$.*

Proof. We define a recursive chain B , and then show that $A = P - B$ can be recursively 5-covered. Let

b_0 = the $\mathbb{N}$ -least element of P .

b_{i+1} = the $\mathbb{N}$ -least x such that $b_i <^{\mathbb{N}} x$ and x is P -comparable to $b_1, \dots, b_i$ (if no such x exists, then $b_{i+1} = b_i$).

$B = \{b_i \mid i \in \mathbb{N}\}$

The set B might be finite, but note that B is recursive and that an algorithm for it can be obtained effectively from an index for $\langle P, \leq^P \rangle$.

Note that B is a recursive chain and that for all $p \in A$ there exists $p' \in B$ such that $p' <^N p$ and $p \mid p'$ (we will use this later).

By convention, elements of A will be denoted by small letters (e.g., p), elements of B will be denoted by small letters with primes (e.g. p'). Usually p and p' will be P -incomparable elements.

To show that A is recursively 5-coverable we will define a total ordering $\leq^*$ and an equivalence relation $\sim$ such that the following hold (the class that p is in is denoted $[p]$).

- (0a) Every equivalence class of $\sim$ is a $\leq^P$ -chain.
- (0b) If $p <^* q <^* r$ and $p \sim r$, then $p \sim q \sim r$.
- (0c) If $x_1 \sim x_2$, $y_1 \sim y_2$, and $x_1 \approx y_1$, then $x_1 <^* y_1$ iff $x_2 <^* y_2$. Hence we can define $<^*$ and $\leq^*$ on equivalence classes via $[p] <^* [q]$ iff $[p] \neq [q]$ and $p <^* q$; and via $[p] \leq^* [q]$ iff $[p] = [q]$ or $[p] <^* [q]$. Both these definitions are independent of the representatives from $[p]$ or $[q]$ that are chosen.
- (0d) If $[p] <^* [q] <^* [r] <^* [s]$, then $p <^P s$.
- (0e) Both $\leq^*$ and $\sim$ are recursive.

We postpone the definitions of $\leq^*$ and $\sim$.

Notation 6.20 If $A \subseteq \mathbb{N}$ and $n \in \mathbb{N}$, then A^n is the set containing the first n elements of A numerically.

Claim 0. *If $\leq^*$ and $\sim$ can be defined to satisfy (0a)–(0e), then there is a recursive 5-covering of A .*

Proof of Claim 0. We describe a recursive 5-covering of A . Inductively assume that the elements of A^n have been distributed among 5 disjoint sets $C_1, \dots, C_5$ such that

- (1) if $p \sim q$, then p and q are in the same C_i (hence we may speak of the set that $[p]$ is in),

- (2) if $p \approx s$ and $p, s \in C_i$, then there exist q and r such that either $[p] <^* [q] <^* [r] <^* [s]$, or $[s] <^* [r] <^* [q] <^* [p]$, and
- (3) each C_i is a $\leq^P$ -chain, this follows from (2) and condition (0d).

Let a be the numerically $(n+1)$ -st element of A . We determine which set to place a into by going through the following cases in order. (All elements referred to below, except a , are in A^n .)

CASE 1: There exists $p \in C_i$ such that $p \sim a$. Then place a into C_i . It is easy to see that the inductive conditions still hold.

CASE 2: There exists an i such that $C_i = \emptyset$. Let i_0 be the least such i . Place a into C_{i_0} . It is easy to see that the inductive conditions still hold.

CASE 3: There exist r, s and t such that $[a] <^* [r] <^* [s] <^* [t]$, $t \in C_i$, and for no $t' \in C_i$ is $[t'] <^* [t]$. Then place a into set C_i . It is easy to see that the inductive conditions still hold.

CASE 4: There exist o, p and q such that $[o] <^* [p] <^* [q] <^* [a]$, $o \in C_i$, and for no $o' \in C_i$ is $[o] <^* [o']$. Similar to Case 3.

CASE 5: There exist elements o, p, q, r, s and t such that $[o] <^* [p] <^* [q] <^* [a] <^* [r] <^* [s] <^* [t]$, and there exists i such that $o, t \in C_i$, and for all u with $[o] <^* [u] <^* [t]$, $u \notin C_i$. Place a into C_i . By (0d) and the induction hypothesis, the inductive conditions still hold.

We show that at least one of these cases occurs. In particular, we assume that none of Cases 1, 2, 3, or 4 occur, and show that Case 5 holds. Let k (or m) be the number of sets containing elements that are $*$ -smaller (or $*$ -larger) than a . Let $C_{i_1}, \dots, C_{i_k}$ (or $C_{j_1}, \dots, C_{j_m}$) be all the sets containing elements that are $*$ -smaller (or $*$ -larger) than a . Let $b_{i_1}, \dots, b_{i_k}$ (or $b_{j_1}, \dots, b_{j_m}$) be the $*$ -largest (or $*$ -smallest) element of $C_{i_1}, \dots, C_{i_k}$ (or $C_{j_1}, \dots, C_{j_m}$) that is $*$ -smaller than a (or $*$ -larger than a). Without loss of generality assume the following holds:

$$[b_{i_k}] <^* \dots <^* [b_{i_1}] <^* [a] <^* [b_{j_1}] <^* \dots <^* [b_{j_m}].$$

We show that $k \geq 2$ (a similar proof shows $m \geq 2$). Assume, by way of contradiction, that $k \leq 1$. Since Case 2 does not hold, $m \geq 4$. Hence there exist three elements $r, s, t \in \{b_{j_1}, b_{j_2}, b_{j_3}, b_{j_4}\}$ such that $r <^* s <^* t$ and, if $k = 1$, $r, s, t \notin [b_{i_1}]$. Case 3 holds with these values of r, s and t , which is a contradiction.

Let $p = b_{i_2}$, $q = b_{i_1}$, $r = b_{j_1}$ and $s = b_{j_2}$. Let o be some element that has been placed into a set, but $o \notin C_{i_1} \cup C_{i_2} \cup C_{j_1} \cup C_{j_2}$ (such exists, since by the negation of Case 2 all five sets are used). Assume $o <^* a$ (the case $a <^* o$ is similar, though there we would call the element t instead of o). Since $o \notin [p] \cup [q]$ we have $[o] <^* [p] <^* [q] <^* [a]$. Let C be the set that o is in. We can assume that o is the $*$ -largest such element of C that is $<^* a$. C must also contain some element t , $a <^* t$, else Case 4 holds. Let t be the $*$ -least such element. Since $t \in C$, $t \notin [r] \cup [s]$. Hence $[a] <^* [r] <^* [s] <^* [t]$. Since o is the $*$ -largest element of C that is $\leq^* a$, and t is the $*$ -smallest element of C that is $\geq^* a$, the elements o, p, q, r, s and t satisfy Case 5. $\square$

We describe a relation $\leq^*$ on A and then prove that it is a recursive linear ordering.

Definition 6.21 If $p \in A$, then $\text{inc}(p) = \{p' : p' \in B \text{ and } p' \mid p\}$. If $p, q \in A$ then $\text{inc}(p) <^P \text{inc}(q)$ means that $(\forall p' \in \text{inc}(p))(\forall q' \in \text{inc}(q)) (p' <^P q')$. Let $\text{inc}(p) \leq^P \text{inc}(q)$ mean that either $\text{inc}(p) <^P \text{inc}(q)$ or $\text{inc}(p) = \text{inc}(q)$. If $\text{inc}(p) \leq^P \text{inc}(q)$ or $\text{inc}(q) \leq^P \text{inc}(p)$, then $\text{inc}(p)$ and $\text{inc}(q)$ are P -comparable.

Definition 6.22 We define a relation $\leq^*$ on A . We later show that $\leq^*$ is a recursive linear ordering. Given $p, q \in A$ apply the least case below that is satisfied by p and q .

- (i) if $p \leq^P q$, then $p \leq^* q$,
- (ii) if $q \leq^P p$, then $q \leq^* p$,
- (iii) if $\text{inc}(p) <^P \text{inc}(q)$ then $p \leq^* q$,
- (iv) if $\text{inc}(q) <^P \text{inc}(p)$ then $q \leq^* p$.
- (v) if none of the above cases apply then p, q are $*$ -incomparable (we later show this never occurs).

Claim 1.

- (1a) For every $p, q \in A$, either $p \leq^* q$ or $q \leq^* p$.
- (1b) $\leq^*$ is reflexive and transitive.
- (1c) $\leq^*$ is a recursive linear ordering.
- (1d) If $p, q, r \in A$, $p \leq^* q \leq^* r$, and $s' \in \text{inc}(p) \cap \text{inc}(r)$, then $s' \in \text{inc}(q)$.
- (1e) If $p <^* q$ and $\text{inc}(p) \cap \text{inc}(q) \neq \emptyset$, then $p <^P q$.

Proof of Claim 1.

(1a): We show that if $p \mid q$ then either $\text{inc}(p) <^P \text{inc}(q)$ or $\text{inc}(q) <^P \text{inc}(p)$ (which implies that p and q are $*$ -comparable). If not, then there exist $p', p'' \in \text{inc}(p)$ and $q' \in \text{inc}(q)$ such that $p' <^P q' <^P p''$ (or the analogue with q' , p' and q''). Since P has width 2, and $p \mid q$ and $q' \mid q$, we have that p is comparable to q' . However, $p <^P q'$ yields $p <^P p''$, and $q' <^P p$ yields $p' <^P p$, both of which contradict $p', p'' \in \text{inc}(p)$.

(1b): $\leq^*$ is clearly reflexive. We show that $\leq^*$ is transitive. Assume $p \leq^* q$ and $q \leq^* r$ and that p , q , and r are distinct. There are several cases to consider.

- (i) If $p \leq^P q$ and $q \leq^P r$ then $p \leq^P r$, hence $p \leq^* r$.
- (ii) If $\text{inc}(p) <^P \text{inc}(q)$ and $\text{inc}(q) <^P \text{inc}(r)$, then $\text{inc}(p) <^P \text{inc}(r)$, hence $p \leq^* r$.
- (iii) Assume $p \leq^P q$ and $\text{inc}(q) \leq^P \text{inc}(r)$. By (1a) either $p \leq^* r$ or $r \leq^* p$. Assume, by way of contradiction, that $r \leq^* p$. If $r \leq^P p$, then $r \leq^P p \leq^P q$, contradicting $q \leq^* r$. If $\text{inc}(r) <^P \text{inc}(p)$, then $\text{inc}(q) <^P \text{inc}(p)$. Hence we have $p <^P q$, $q' <^P p'$ (for all $p' \in \text{inc}(p)$, and $q' \in \text{inc}(q)$). The elements p and q' must be P -comparable, since otherwise $q' \in \text{inc}(p)$, which violates $\text{inc}(q) <^P \text{inc}(p)$. But $p \leq^P q'$ implies $p \leq^P p'$, and $q' \leq^P p$ implies $q' \leq^P q$, both of which are contradictions. Hence $p \leq^* r$.
- (iv) Assume $\text{inc}(p) < \text{inc}(q)$ and $q \leq^P r$. Similar to (iii).

(1c): From (1a) and (1b), $\leq^*$ is a linear ordering. We describe an algorithm that determines how p and q $*$ -compare. First determine how p and q P -compare. If $p \leq^P q$ then $p \leq^* q$, and if $q \leq^P p$ then $q \leq^* p$. If $p \nmid q$, then by (1a), $\text{inc}(p)$ and $\text{inc}(q)$ P -compare. Find $p' \in \text{inc}(p)$ and $q' \in \text{inc}(q)$. Now $p <^* q$ iff $p' <^P q'$.

(1d): Assume by way of contradiction that $s' <^P q$. Then $q \not\leq^P r$ (else $s' <^P r$). Since $q <^* r$ and $q \not\leq^P r$, we have $\text{inc}(q) <^P \text{inc}(r)$. Since $s' \in \text{inc}(r)$, we have $\text{inc}(q) <^P s' <^P q$, which is a contradiction. Hence $s' \not\leq^P q$. Similar reasoning yields $q \not\leq^P s'$, so $s' \in \text{inc}(q)$.

(1e): Let $x \in \text{inc}(p) \cap \text{inc}(q)$. Since P has width 2 and $p \mid x, q \mid x$ we know p and q are P -comparable. If $q <^P p$, then $q \leq^* p$, hence $p <^P q$. $\square$

We describe a recursive equivalence relation $\sim$ on A inductively. Assume that the elements of A^n have been put into equivalence classes. Given q , the numerically $(n+1)$ -st element of A , we proceed as follows. Find $q^-, q^+ \in A^n$ (if they exist) such that

q^- is the $*$ -max element such that $q^- <^* q$.

q^+ is the $*$ -min element such that $q <^* q^+$.

(Note that $q^- <^* q <^* q^+$.)

If there exists $q' \in B$, $q' <^N q$, $q' \mid q$ and $q' \mid q^-$, then place q in the same class as q^- . If not, but if there exists $q' <^N q$, $q' \mid q$ and $q' \mid q^+$, then place q in the same class as q^+ . If neither of these occurs, then q becomes the first element of a new class.

We can now prove (0a)–(0c), the first three properties that were required of $\leq^*$ and $\sim$. We restate them because we need a slightly stronger version (strengthening the induction hypothesis).

Claim 2. *For every n , when only the elements of A^n are put into classes, the following hold.*

(2a) *If p and q are two $*$ -adjacent elements such that $[p] = [q]$, then*

(1) $\exists x \in \text{inc}(p) \cap \text{inc}(q)$ with $x <^N \mathbf{N}\text{-max}\{p, q\}$, and

(2) p and q are P -comparable. Every class is a $\leq^P$ -chain.

(2b) If $p <^* q <^* r$ and $p \sim r$, then $p \sim q \sim r$.

(2c) If $p_1 \sim p_2$, $q_1 \sim q_2$, and $p_1 \approx q_1$, then $p_1 <^* q_1$ iff $p_2 <^* q_2$.

Proof of Claim 2. By induction on n . Assume all three items are true for A^n , and consider what may happen when q , the numerically $(n+1)$ -st element of A , is considered. Let q^- and q^+ be as in the definition of $\sim$. Note that q^- and q^+ are $*$ -adjacent elements in A^n .

CASE 1: If $q^- \sim q^+$, then q^- and q^+ are $*$ -adjacent elements, and $[q^-] = [q^+]$. By the induction hypothesis there exists $x \in \text{inc}(q^-) \cap \text{inc}(q^+)$ with $x <^N \text{N-max}\{q^-, q^+\}$. By Claim (1d), we have $x \in \text{inc}(q)$. Since $x <^N \text{N-max}\{q^-, q^+\} <^N q$, the element q is placed in $[q^-]$. By Claim (1e), $q^- \leq^P q \leq^P q^+$. Hence (2a) holds. It is easy to see that (2b) and (2c) hold as well.

CASE 2: $q^- \approx q^+$ and $q^- \sim q$. Since q was placed in $[q^-]$, we have $(\exists x <^N q)[x \in \text{inc}(q) \cap \text{inc}(q^-)]$. By Claim (1e), $q^- \leq^P q$. Hence, since $q = \text{N-max}\{q, q^-\}$, (2a) holds. It is easy to see that (2b) and (2c) hold as well.

CASE 3: $q^- \approx q^+$ and $q^+ \sim q$. Similar to Case 2.

CASE 4: q becomes the first element of a new class. In this case (2a), (2b) and (2c) hold trivially. $\square$

Definition 6.23 We define $<^*$ and $\leq^*$ on classes via $[p] <^* [q]$ iff $[p] \neq [q]$ and $p <^* q$; and $[p] \leq^* [q]$ iff $[p] = [q]$ or $[p] <^* [q]$. Claim (2c) shows that these definitions are independent of representation.

We need one more claim before we can prove item (0d) about $\sim$.

Claim 3. Let a be the N -least element of $[a]$. Let b be such that $a <^* b$.

- (i) If there exists c' such that $c' \mid a$, $c' \mid b$, $c' <^N a$ and $c' \in B$, then $a \sim b$.
- (ii) If $a \approx b$, then for all $c' <^N a$ such that $c' \in B$ and $c' \mid a$, we have $c' <^P b$.
- (iii) If $a \approx b$, then there exists d such that $a \approx d$, $[a] <^* [d] \leq^* [b]$, d is the N -least element of $[d]$, and for all c' such that $c' <^N d$, $c' \in B$, and $c' \mid d$, we have $a <^P c'$.

Proof of Claim 3.

(i): We first show that $a <^N b$. Let $S = \{d \in A : d <^N a, a <^* d \leq^* b\}$. We show $S = \emptyset$ which easily implies $a <^N b$. Assume, by way of contradiction, that $S \neq \emptyset$. Let d be the $*$ -smallest element of S . Note that

- (1) since $a <^* d \leq^* b$, $c' \mid a$, and $c' \mid b$, by Claim (1d), $c' \mid d$, and
- (2) when a is placed into an equivalence class, d is the value of a^+ (i.e., the $*$ -least element that is $*$ -larger than a and N -less than a).

Consider what happens when a is placed into an equivalence class. If a is placed into $[a^-]$ then a will not be the N -least element of $[a]$. If a is not placed into $[a^-]$ then, since $c' \mid a$, $c' \mid d$, $c' <^N a$, and $d = a^+$, a is placed into $[d]$. In either case a is not the N -least element of its class, contrary to hypothesis. Hence $S = \emptyset$.

We now know that when b is placed into a class, a has already been so placed. We show $b \sim a$ by induction on n , the number of elements N -smaller than b and $*$ -between a and b when b is considered. If $n = 0$, then when b is considered $a <^* b$ (adjacent), $c' \mid a$, $c' \mid b$, and $c' <^N a <^N b$. Hence, b will be placed in $[a]$. If $n > 0$, then let b^- be the $*$ -largest element that is $*$ -less than b when b is placed (so $a <^* \dots <^* b^- <^* b$). Since $c' \mid a$ and $c' \mid b$, by Claim (1d) $c' \mid b^-$. Since there are $\leq^N n - 1$ elements $*$ -between a and b^- when b^- is placed, and since $c' <^N a <^N b^-$ (by the above argument) we may apply the induction hypothesis to a and b^- , hence $a \sim b^-$. Thus b will be placed into $[b^-] = [a]$, so $a \sim b$.

(ii): By the contrapositive of (i), we know that c' is comparable to b . Since $a <^* b$, either $a <^P b$ or $\text{inc}(a) <^P \text{inc}(b)$. Either case leads to $b \not\leq^P c'$, hence $c' <^P b$.

(iii): If $b <^N a$, let d be the N -least element of $[b]$ (we prove later that this choice of d works). If $a <^N b$, let $S = \{d \in A : d \leq^N b, a <^* d \leq^* b\}$. Let d^* be the $*$ -least element of S such that $a \approx d^*$. (Note that d^* exists, since $b \in S$ and $a \approx b$.) Let d be the N -least element of $[d^*]$.

The remainder of this proof is valid for either choice of d . Since $a \sim a$, $d \sim d^*$, $a \approx d^*$, and $a <^* d$, by Claim (2c) $[a] <^* [d]$. Since $d \leq^* b$, by definition $[d] \leq^* [b]$. Hence $[a] <^* [d] \leq^* [b]$.

Let $c' \mid d$, $c' <^N d$, and $c' \in B$. We first show that c' and a are P -comparable, and second that $a <^P c'$. There are two cases.

CASE 1: $c' <^N a$. Note that $c' \mid d$, $c' <^N a$, and $c' \in B$. If $c' \mid a$, then by (i) (of this claim), $a \sim d$. Hence c' and a are P -comparable.

CASE 2: $a <^N c' <^N d$. Note that $a <^N d$. Hence when d is placed into a class, a has already been so placed. Let d^- be as in the definition of classes. Note that, since d is the N -least element of $[d]$, we have $[a] \leq^* [d^-] <^* [d]$. Note that $c' \mid d$. If $c' \mid a$, then by Claim (1d), $c' \mid d^-$. But then d will be placed into $[d^-]$, which is a contradiction.

We show that $a <^P c'$. Since $a <^* d$ either $a \leq^P d$ or $\text{inc}(a) <^P \text{inc}(d)$. Either case implies $c' \not\leq^P a$. Since a, c' are P -comparable we have $a <^P c'$. $\square$

We can now prove (0d).

Claim 4. If $[p] <^* [q] <^* [r] <^* [s]$, then $p <^P s$.

Proof of Claim 4. We can take q to be the N -least element of $[q]$. Let d be as in Claim 3(iii) with $a = q$ and $b = r$, and let $q', d' \in B$ be such that $q \mid q'$, $q' <^N q$, $d \mid d'$, and $d' <^N d$. Then $[p] <^* [q] <^* [d] \leq^* [r] <^* [s]$, d is the N -least element of $[d]$, and $q <^P d'$. By Claim 3(ii) with $a = d$, $b = s$ and $c' = d'$, we obtain $d' <^P s$. Since $q <^P d'$, and q', d' are comparable (as all elements of B are), $q' <^P d'$ (else $q <^P d' \leq^P q'$). If $p <^P q$, then we have $p <^P q <^P d' <^P s$, so we are done. If $p \mid q$, then since $q \mid q'$ and $\langle P, \leq^P \rangle$ has width 2, p is comparable to q' . Assume, by way of contradiction, that $q' <^P p$. Since $p <^* q$, either

(1) $p <^P q$, so $q' <^P p <^P q$, or

(2) $\text{inc}(p) <^* \text{inc}(q)$, so $p' <^P q' <^P p$ for $p' \in \text{inc}(p)$.

Hence we have $p <^P q'$, so $p <^P q' <^P d' <^P s$. $\square$ $\square$

Theorem 6.24 If $\langle P, \leq^P \rangle$ is a recursive partial order of width w , then P has recursive width $\leq \frac{1}{4}(5^w - 1)$. Moreover, given an index for $\langle P, \leq^P \rangle$, one can recursively find an index for a recursive $\frac{1}{4}(5^w - 1)$ -covering.

Sketch Proof. This is a proof by induction. The base case of $w = 2$ is Theorem 6.19. Let $\langle P, \leq^P \rangle$ be a partial order of width $w \geq 3$. First, a linear suborder is defined similar to B in Theorem 6.19. Second, a recursive partial order $\leq^*$ on $A = P - B$ is defined which is somewhat similar to the $\leq^*$

in Theorem 6.19. Third, prove that $\langle A, \leq^* \rangle$ is a recursive partial order of width $w - 1$. By induction, $\langle A, \leq^* \rangle$ has recursive width $\leq \frac{1}{4}(5^{w-1} - 1)$. We then show that every recursive $*$ -chain of A can be covered by five P -chains. Thus P can be recursively covered by $1 + 5(\frac{1}{4}(5^{w-1} - 1)) = \frac{1}{4}(5^w - 1)$ recursive chains. $\square$

6.4.2 Bounding the recursive width given partial information

In the algorithms in Theorems 6.19 and 6.24 we needed the ability to tell *how* elements compared. What happens if we only have the ability to tell *if* elements compare?

Definition 6.25 If $\mathcal{P} = \langle P, \leq \rangle$ is a partial order, then the *co-comparability graph* of $\mathcal{P}$ is $G_{\mathcal{P}} = (P, E)$, where $E = \{\{x, y\} : x, y \text{ are incomparable}\}$. The set of all co-comparability graphs of finite or countable partial orders is denoted Γ_{co} .

Notation 6.26 If G is a graph, then $\omega(G)$ is the size of the largest clique in G .

A (recursive) a -coloring of $G_{\mathcal{P}}$ yields a (recursive) a -covering of $\mathcal{P}$; and $w(\mathcal{P}) = \omega(G_{\mathcal{P}})$. Hence we examine recursive colorings of recursive graphs in $G \in \Gamma_{\text{co}}$ with fixed $\omega(G)$.

Schmerl asked if there exists a function f such that, for every recursive $G \in \Gamma_{\text{co}}$, $\chi^r(G) \leq f(\omega(G))$. Kierstead, Penrice, and Trotter [101] answered this affirmatively. Their result is a corollary of a theorem in combinatorics, and is part of a fascinating line of research. We sketch that line of research and their theorem. For a fuller account of this area see [98].

Definition 6.27 A class of graphs $\mathcal{G}$ is χ -bounded if there exists a function f such that, for all $G \in \mathcal{G}$, $\chi(G) \leq f(\omega(G))$. (The function f need not be computable.)

Notation 6.28 If H is a graph, then $\text{Forb}(H)$ is the set of graphs that do not contain H as an induced subgraph. (“*Forb*” stands for “Forbidden”.)

The question arises as to which classes of graphs are χ -bounded. If $\text{Forb}(H)$ is χ -bounded then, by a result of Erdős and Hajnal [52], H is

acyclic. Gyárfás [70] and Sumner [162] conjectured the converse, i.e., if T is a tree, then the class $\text{Forb}(T)$ is χ -bounded. Gyárfás [71] showed that if P is a path, i.e., a graph with

$$V = \{v_1, \dots, v_k\}$$

and

$$E = \{\{v_i, v_{i+1}\} : i = 1, \dots, k-1\},$$

then $\text{Forb}(P)$ is χ -bounded. Kierstead and Penrice [100] showed that if T is a tree of radius 2 (i.e., there is a vertex v such that for all vertices x there is a path of length ≤ 2 from v to x), then $\text{Forb}(T)$ is χ -bounded. More is known if we restrict attention to on-line colorings.

Definition 6.29 An *on-line graph* is a structure $G^< = (V, E, <)$ where $G = (V, E)$ is a graph and $<$ is a linear ordering of V (if V is infinite then $<$ has the order type of the natural numbers). $G^<$ is an *on-line presentation* of G .

Definition 6.30 Let $G^< = (V, E, <)$ and $V = \{v_1 < v_2 < \dots\}$. An *on-line algorithm to color $G^<$* is an algorithm that colors $v_1, v_2, \dots$ in order, so that the color assigned to v_i depends only on G restricted to $\{v_1, \dots, v_i\}$.

Definition 6.31 A class of graphs $\mathcal{G}$ is *on-line χ -bounded* if there exists a function f such that, for all $G \in \mathcal{G}$ and for all on-line presentations $G^<$ of G , there exists an on-line algorithm to color $G^<$ with $\leq f(\omega(G))$ colors.

The question arises as to which classes of graphs are on-line χ -bounded. Chvátal [40] showed that $\text{Forb}(P_4)$ is on-line χ -bounded where P_n is the path on n vertices. Gyárfás and Lehel [72] showed that $\text{Forb}(P_5)$ is on-line χ -bounded, but that $\text{Forb}(P_6)$ is not. Hence if $\text{Forb}(T)$ is on-line χ -bounded for some tree T , then T has radius 2. Kierstead, Penrice, and Trotter [101] proved this condition is not only necessary but also sufficient. Combining the results we have mentioned of Erdős-Hajnal, Gyárfás-Lehel (that $\text{Forb}(P_6)$ is not χ -bounded), and Kierstead-Penrice-Trotter, one obtains the following.

Theorem 6.32 *Let G be a connected graph. $\text{Forb}(G)$ is on-line χ -bounded iff G is a tree of radius 2.*

Theorem 6.32 can be applied to Γ_{co} .

Corollary 6.33 Γ_{co} is on-line χ -bounded.

Proof. Let T be the tree formed by subdividing each edge of $K_{1,3}$ (i.e., place a new vertex on every edge of $K_{1,3}$). By a case analysis, one can verify that $\Gamma_{co} \subseteq \text{Forb}(T)$. Since T has radius 2, by Theorem 6.32, $\text{Forb}(T)$ is on-line χ -bounded. $\square$

Every recursive $G \in \Gamma_{co}$ has a recursive on-line presentation. Hence the on-line coloring algorithm in Corollary 6.33 yields a recursive coloring of all recursive graphs in Γ_{co} . Hence we have the following corollaries.

Corollary 6.34 *There exists a function f such that*

- (i) *for every recursive $G \in \Gamma_{co}$, $\chi^r(G) \leq f(\omega(G))$, and*
- (ii) *for every partial order $\mathcal{P}$, $w^r(\mathcal{P}) \leq f(w(\mathcal{P}))$ via an algorithm that only uses the information in $G_{\mathcal{P}}$.*

If we did not already have Theorem 6.24, then we could have used Corollary 6.34 to obtain *some* bound on $w^r(\mathcal{P})$ in terms of $w(\mathcal{P})$. The function $f(w)$ obtained in the proof of Corollary 6.34 is rather complicated, and grows faster than $\frac{1}{4}(5^w - 1)$, though it is bounded by an exponential. Hence it does not offer an improvement to the bound in Theorem 6.24. However, since the recursive covering uses less information, it is an improvement in that sense.

6.5 Recursion-theoretic modification

By Theorem 6.14 there are recursive partial orders of width w that are not recursively w -coverable. We prove that there is always a w -covering of low degree.

Theorem 6.35 *If $\langle P, \leq \rangle$ is a recursive partial order of width w , then there exists a w -covering of low degree.*

Proof. Assume, without loss of generality, that $P = \mathbb{N}$ (but of course $\leq$ has no relation to $\leq^{\mathbb{N}}$). Consider the following recursive w -ary tree: The vertex $\sigma = (a_1, \dots, a_n)$ is on T iff

- (i) For all i , $1 \leq i \leq n$ we have $1 \leq a_i \leq w$.
- (ii) the map that sends i to a_i is a w -covering of $\langle P, \leq \rangle$ restricted to $\{1, \dots, n\}$.

We have

- (1) T is recursive,
- (2) T is recursively bounded by the function $f(n) = \langle w, \dots, w \rangle$ (w appears n times),
- (3) any infinite branch T is a w -covering of $\langle P, \leq \rangle$,
- (4) every w -covering of $\langle P, \leq \rangle$ is represented by some infinite branch of T , and
- (5) the set of infinite branches of T is nonempty (by the classical Dilworth's Theorem and the previous item).

Since the branches of T form a nonempty Π_1^0 class, by Theorem 2.12 there exists an infinite low branch. This branch represents a covering of low degree. $\square$

6.6 Recursion-combinatorial modification

We now consider an ‘effective version’ of Dilworth’s theorem which is true. The modification is both recursion-theoretic and combinatorial. It is not quite as effective as we might like: while it shows that (under certain conditions) a partial order of width w has a recursive w -covering, the proof is not uniform. Schmerl [147] showed that the proof cannot be made uniform.

This effective version is reported without proof in [94] and credited to Schmerl. This is the first published account.

Definition 6.36 Let $\langle P, \leq \rangle$ be a partial order. Define $I : P \times P \rightarrow 2^P$ via $I(x, y) = \{z \in P \mid x \leq z \leq y\}$. (I stands for “In between”.)

Definition 6.37 Let $\langle P, \leq \rangle$ be a partial order. $\langle P, \leq \rangle$ is *locally finite* if for all x, y , the set $I(x, y)$ is finite. $\langle P, \leq \rangle$ is *recursively locally finite* if it is a locally finite recursive partial order, and the function I is recursive. We abbreviate “recursive locally finite partial order” by “r.l.f.p.o.”, and “recursively locally finite” by “r.l.f.”. Indices for r.l.f.p.o.’s can easily be defined.

Note that the above definition is equivalent to being able to recursively find $|I(x, y)|$.

Theorem 6.38 *If $\mathcal{P} = \langle P, \leq \rangle$ is an r.l.f.p.o., then $w(\mathcal{P}) = w^r(\mathcal{P})$.*

Proof. We prove this theorem by induction on $w = w(\mathcal{P})$. For $w = 1$ the theorem is trivial. Assume it holds for all $\mathcal{P}$ such that $w(\mathcal{P}) \leq w - 1$. Let $\mathcal{P} = \langle P, \leq \rangle$ be a recursive locally finite partial order of width w . Let $\{a_1, \dots, a_w\}$ be a w -antichain in $\mathcal{P}$. (One cause of the proof being nonuniform is that we need to find a w -antichain. If one allows w as a parameter to an alleged uniform algorithm then w then this step would not cause non-uniformity.) Let

$$P_1 = P \cap \{x \mid (\exists i) x \geq a_i\}$$

$$P_2 = P \cap \{x \mid (\exists i) x \leq a_i\}$$

Let $\mathcal{P}_1 = \langle P_1, \leq \rangle$ and $\mathcal{P}_2 = \langle P_2, \leq \rangle$. We will recursively w -cover $\mathcal{P}_1$ and $\mathcal{P}_2$, and then combine these w -coverings into a recursive w -covering of $\mathcal{P}$ (this combining is easy and hence omitted). The advantage of working with $\mathcal{P}_1$ (or $\mathcal{P}_2$) instead of $\mathcal{P}$ is that $\mathcal{P}_1$ (or $\mathcal{P}_2$) has no infinite descending (ascending) chains.

We describe how to recursively w -cover $\mathcal{P}_1$. The w -covering of $\mathcal{P}_2$ is similar.

Let $\mathcal{A}$ be the set of all w -antichains of $\mathcal{P}_1$ (note that $\mathcal{A}$, suitably coded, is recursive). A chain C is *saturated* if for every $A \in \mathcal{A}$, $A \cap C \neq \emptyset$ (since A is an antichain and C is a chain, $|A \cap C| = 1$). It is clear that if C is a recursive chain which is saturated, then $\mathcal{P}' = \langle P_1 - C, \leq \rangle$ is r.l.f., and $w(\mathcal{P}') = w - 1$. Hence it suffices to construct a recursive chain C which is saturated, and then use the induction hypothesis on $\mathcal{P}'$.

We define a recursive partial order on $\mathcal{A}$ as follows:

$$A \leq B \quad \text{iff} \quad (\forall a \in A)(\exists b \in B)[a \leq b]$$

(as usual $A < B$ means $A \leq B$ and $A \neq B$). We define a binary operation glb on pairs of elements of $\mathcal{A}$. We will later see that $glb(A, B) \leq A, B$, and no antichain that is larger has this property (so $glb(A, B)$ is the greatest lower bound of $\{A, B\}$). If $A, B \in \mathcal{A}$ then

$$glb(A, B) = \{x \in A \cup B : (\forall y \in A \cup B)[x \text{ comparable to } y \Rightarrow x \leq y]\}.$$

We show that $glb(A, B) \in \mathcal{A}$, i.e., $glb(A, B)$ is an antichain of size w . Clearly $glb(A, B)$ is an antichain. Hence $|glb(A, B)| \leq w$. We show that

$|glb(A, B)| \geq w$. Let $\overline{glb(A, B)}$ denote $(A \cup B) - glb(A, B)$. Note that $A \cap B \subseteq glb(A, B)$, and that $(A \cap B) \cup \overline{glb(A, B)}$ is an antichain, so it has $\leq w$ elements. Using this we have the following.

$$|glb(A, B)| + |\overline{glb(A, B)}| = |A \cup B|$$

$$|glb(A, B)| + |\overline{glb(A, B)}| = 2w - |A \cap B|$$

$$|glb(A, B)| = 2w - |A \cap B| - |\overline{glb(A, B)}|$$

$$|glb(A, B)| = 2w - |(A \cap B) \cup \overline{glb(A, B)}|$$

(above line uses $A \cap B \subseteq glb(A, B)$)

$$|glb(A, B)| \geq 2w - w = w \quad (\text{use } |(A \cap B) \cup \overline{glb(A, B)}| \leq w).$$

Using that $\mathcal{P}_1$ is r.l.f. and has no infinite descending chains, one can show the following.

- (i) Let $A, B \in \mathcal{A}$. $glb(A, B) \leq A, B$. For all D , if $D \leq A, B$, then $D \leq glb(A, B)$. (I.e., $glb(A, B)$ is the greatest lower bound of A, B .)
- (ii) Let $B \in \mathcal{A}$. Let $LESS(B) = \{X \in \mathcal{A} \mid X \leq B\}$. This set is finite. Moreover, given $B \in \mathcal{A}$ one can effectively find $LESS(B)$, suitably coded.
- (iii) Let $B \in \mathcal{A}$. The set $LESS(B)$ has a *unique* minimal element. That is, there exists *exactly one* antichain $A \in LESS(B)$ such that for all $D \in LESS(B)$, $D \not\leq A$. (Proof: If A and D are both minimal, then note $glb(A, D) \leq A, D$ and $glb(A, D) \in LESS(B)$.) One can easily find the minimal antichain by brute force and (ii).
- (iv) Let $\mathcal{A}' \subseteq \mathcal{A}$ be such that if $A, D \in \mathcal{A}'$ then $glb(A, D) \in \mathcal{A}'$. There exists a unique minimal antichain in $\mathcal{A}'$. Moreover, given an index for $\mathcal{A}'$, one can effectively find that antichain (unless $\mathcal{A}' = \emptyset$ in which case the algorithm diverges). To do this, find some $B \in \mathcal{A}'$ and then use (iii). (Proof of uniqueness: if A, D are both minimal, then note $glb(A, D) \leq A, D$ and $glb(A, D) \in \mathcal{A}'$.)

Using item (iv), the following function from $2^{\mathcal{A}}$ to $\mathcal{A}$ is well defined and can be partially computed via indices:

$$\text{less}(\mathcal{A}') = \begin{cases} \text{the min. antichain in } \mathcal{A}' & \text{if } \mathcal{A}' \text{ is closed under glb and } \mathcal{A}' \neq \emptyset; \\ \uparrow & \text{otherwise.} \end{cases}$$

If $\text{less}(\mathcal{A}') \downarrow$, then we say that $\text{less}(\mathcal{A}')$ exists.

We construct a recursive saturated chain in stages. The construction might get stuck in some stage forever. If so then the chain constructed will still be saturated, and will be finite (hence recursive). We cannot tell which case occurs, hence we cannot (from this proof) obtain an index for the set of elements in the chain. This is why the proof is nonuniform.

Let C_s denote the finite chain constructed by the end of stage s . Let $\mathcal{A}_s = \{A \in \mathcal{A} \mid A \cap C_s = \emptyset\}$. It is easy to see that $\mathcal{A}_s$ is recursive and that, given C_s and an index for $\mathcal{A}$, we can effectively obtain an index for $\mathcal{A}_s$. For all $s \geq 0$, C_s and $\mathcal{A}_s$ will satisfy the following

- (1) $(\forall A' \in \mathcal{A}) [A' \cap C_s = \emptyset \text{ iff } A' \in \mathcal{A}_s]$,
- (2) $\text{less}(\mathcal{A}_{s-1}) \notin \mathcal{A}_s$, for $s \geq 1$,
- (3) $(\forall a' \in \bigcup \mathcal{A}_s)(\forall c' \in C_s) [a' \mid c' \vee a' > c']$,
- (4) $A, D \in \mathcal{A}_s \Rightarrow \text{glb}(A, D) \in \mathcal{A}_s$ (hence if $\mathcal{A}_s \neq \emptyset$, then $\text{less}(\mathcal{A}_s)$ exists),
- (5) if $\mathcal{A}_s \neq \emptyset$, then $\text{less}(\mathcal{A}_s) \in \mathcal{A}_s$.

Construction

STAGE 0:

$C_0 = \{a_1\}$ (recall that a_1 was defined when $\mathcal{P}_1, \mathcal{P}_2$ were defined), $\mathcal{A}_0 = \mathcal{A} - \{A \mid a_1 \in A\}$. Clearly (1)–(5) hold for $s = 0$.

STAGE $s + 1$:

Find $A = \text{less}(\mathcal{A}_s)$. (If $\mathcal{A}_s \neq \emptyset$, then by (4) and (5) $\text{less}(\mathcal{A}_s)$ exists; if $\mathcal{A}_s = \emptyset$ then this computation will not halt. This is why the proof is nonuniform.) Let c be the $\leq$ -largest element of C_s . By (3), we have $(\forall a \in A) [a \mid c \vee a > c]$. If $(\forall a \in A) [a \mid c]$, then $A \cup \{c\}$ would be a $(w + 1)$ -antichain, so

$$(\exists a \in A) [a > c].$$

Let $C_{s+1} = C_s \cup \{a\}$, and note that $\mathcal{A}_{s+1} = \mathcal{A}_s - \{A' \mid a \in A'\}$. End of the construction.

If the construction gets stuck during stage $s + 1$ then let $C = C_s$, else let $C = \bigcup_s C_s$.

It is clear that (1), (2), (4) and (5) hold at the end of stage $s + 1$. We show by induction on s that (3) holds. Clearly (3) holds for $s = 0$. Assume (3) is true for s but not for $s + 1$. Hence

$$(\exists a' \in \bigcup \mathcal{A}_{s+1})(\exists c' \in C_{s+1})[a' \leq c'].$$

Since $\mathcal{A}_{s+1} \subseteq \mathcal{A}_s$,

$$(\exists a' \in \bigcup \mathcal{A}_s)(\exists c' \in C_{s+1})[a' \leq c'].$$

Let A and a be as in the construction during stage $s + 1$. Note that $a \neq a'$ by the construction. Since (3) holds for s , the value of c' must be in $C_{s+1} - C_s = \{a\}$, hence $c' = a$. So $(\exists a' \in \bigcup \mathcal{A}_s)[a' < a]$. Let $A_1 \in \mathcal{A}_s$ be such that $a' \in A_1$. Let $A_2 = \text{glb}(A, A_1)$. Note that $A_2 \in \mathcal{A}_s$. Since $a, a' \in A \cup A_1$ and $a' < a$, $a \notin A_2$, so $A_2 \neq A$. Hence $A_2 < A$, which contradicts $A = \text{less}(\mathcal{A}_s)$.

We show that C is saturated. We need some auxiliary notions. For $m \geq 1$, let

$$F_m = \{A \in \mathcal{A} : |\text{LESS}(A)| = m\}.$$

We show that, for every $m \geq 1$, F_m is finite. For $m = 1$, note that F_1 consists of the unique minimal element of $\mathcal{A}$, so $|F_1| = 1$ and the claim is true. Let $m \geq 2$. For every $A \in F_m$, there exists $B \in \text{LESS}(A) \subseteq \bigcup_{1 \leq i < m} F_i$ that is right below A , i.e., there is no D such that $B < D < A$ (if not, then $\text{LESS}(A)$ is infinite). Since $\bigcup_{1 \leq i < m} F_i$ is finite (by the induction hypothesis), and the number of A that are right below a particular element of $\mathcal{A}$ is finite (by local finiteness), the number of elements in F_m is finite.

We show, by induction on m , that for every $A \in F_m$, $A \cap C \neq \emptyset$. For $m = 1$ this is clear, since the minimal antichain of $\mathcal{A}$ is $\{a_1, \dots, a_w\}$, which intersects C_0 . Let $m > 1$, and let $A \in F_m$. Let s be the least stage such that all antichains in $\bigcup_{1 \leq i < m} F_i$ intersect C_s (s exists by the induction hypothesis and the finiteness of $\bigcup_{1 \leq i < m} F_i$). If $A \notin \mathcal{A}_s$ then $A \cap C_s \neq \emptyset$, and if $A \in \mathcal{A}_s$ then $A = \text{less}(\mathcal{A}_s)$, so $A \cap C_{s+1} \neq \emptyset$. In either case $A \cap C \neq \emptyset$. $\square$

The question arises as to whether the proof of Theorem 6.38 can be made uniform. Schmerl [147] showed that, in a strong sense, it cannot.

Definition 6.39 A partial order $\mathcal{P} = \langle P, \leq \rangle$ is a *strongly recursively locally finite partial order* (abbreviated s.r.l.f.p.o.) if it is an r.l.f.p.o., and the following functions from P to $\mathbb{N}$ are recursive.

$$up(x) = \begin{cases} 0 & \text{if } |\{y : x \leq y\}| = \infty, \\ |\{y : x \leq y\}| & \text{otherwise.} \end{cases}$$

$$down(x) = \begin{cases} 0 & \text{if } |\{y : y \leq x\}| = \infty, \\ |\{y : y \leq x\}| & \text{otherwise.} \end{cases}$$

An index for an s.r.l.f.p.o. can easily be defined. $\mathcal{P}_e$ is the s.r.l.f.p.o. that is associated to index e .

Schmerl showed that, even if the index of an s.r.l.f.p.o. of width 2 is given, one cannot uniformly find an index for a recursive 2-covering.

Theorem 6.40 *There does not exist an algorithm $\mathcal{A}$ that, on inputting e , an index for a s.r.l.f.p.o. of width 2, will output an index for a 2-covering of $\mathcal{P}_e$.*

Proof. Assume that such an $\mathcal{A}$ exists. We construct an s.r.l.f.p.o. $\mathcal{P}_e$, such that $\mathcal{A}(e)$ is not an index for a recursive 2-covering of $\mathcal{P}_e$. By the recursion theorem, we can assume that the construction may use e , an index for the s.r.l.f.p.o. being constructed. Let $i = \mathcal{A}(e)$.

Construction

STAGE 0:

Initially the base set is $\{0, 1, 2\}$. The elements 0 and 1 are incomparable, and 2 is greater than both 0 and 1. Set $\text{DIAG} = \text{FALSE}$ (we have not diagonalized against $\{i\}$ yet), and $\text{TOP} = 2$.

STAGE $s + 1$:

Place the least unused number u directly above TOP , and then set $\text{TOP} = u$. If $\text{DIAG} = \text{FALSE}$, then run $\{i\}(0)$, $\{i\}(1)$ and $\{i\}(2)$ for s steps. If all three halt, then set $\text{DIAG} = \text{TRUE}$, and do the following: if $\{i\}(0) = \{i\}(2)$, then place the least unused number above 0 and incomparable to everything else, otherwise place the least unused number above 1 and incomparable to everything else. End of the construction.

It is easy to see that the construction yields an s.r.l.f.p.o. of width 2 that has index e , but $\mathcal{A}(e)$ is not a recursive 2-covering of it. $\square$

6.7 Miscellaneous

We have been concerned with the width of a partial order. Other parameters of partial orders (and recursive partial orders) have also been examined. We state several theorems along these lines. No proofs are given.

6.7.1 Recursive dimension

A *realizer* of a partial order $\langle P, \leq^P \rangle$ is a set of linear orders $L_1, \dots, L_d$ such that each one uses P as its base set, and

$$x \leq^P y \quad \text{iff} \quad (\forall i)(x < y \text{ in } L_i).$$

The *dimension* of a partial order is the minimal number of linear orders in a realizer. (An alternative definition of dimension is the least d such that P can be embedded in $\mathbb{Q}^d$, where $\mathbb{Q}$ is the rationals.) The notions of *recursive realizer* and *recursive dimension* can be defined easily.

It is known that the dimension of a partial order is $\leq$ its width. Is this true for recursive dimension and recursive width? Kierstead, McNulty and Trotter [99] have shown that this is false; but for low widths, bounds on the recursive dimension can be obtained. They showed that if $\mathcal{P}$ is a recursive partial order, then

- (1) if $w^r(\mathcal{P}) \leq 2$, the recursive dimension of $\mathcal{P}$ is ≤ 5 ,
- (2) if $w^r(\mathcal{P}) \leq 3$, the recursive dimension of $\mathcal{P}$ is ≤ 6 (this is tight — there exists a recursive partial order $\mathcal{P}$ with $w^r(\mathcal{P}) = 3$ and recursive dimension 6),
- (3) there is a partial order $\mathcal{P}$ with $w^r(\mathcal{P}) = 4$ which has no finite recursive dimension ($\mathcal{P}$ also has width 3).

If we impose conditions on $\mathcal{P}$, then better bounds can be obtained. Let Q be the order on four elements $\{a, b, c, d\}$ where $a < b, c < d$, and no other pairs of elements are comparable. An *interval order*³ is an order that does not have Q as an induced suborder (alternatively, an interval order is formed by taking the base set to be a set $\{I_1, I_2, \dots\}$ of open intervals of reals, and declaring $I_i < I_j$ iff every element in I_i is less than every element in I_j). Hopkins [83] showed that if $\mathcal{P}$ is a recursive interval order of width w then

³Interval Orders were named by Fishburn in [54] but were known to Norbert Wiener.

- (1) if $w = 2$, $\mathcal{P}$ has recursive dimension ≤ 3 ,
- (2) $\mathcal{P}$ has recursive dimension $\leq 4w - 4$; however, for all $w \geq 2$, there exists a recursive interval order $\mathcal{P}$ of width w that has recursive dimension $\lceil \frac{4}{3}w \rceil$.

If the recursive width is bounded, then there are different results. Kierstead et al. [99] have shown that, for recursive interval orders, if the recursive width is $\leq w$ then the recursive dimension is $\leq 2w$.

A *crown* is a partial order on $\{a_1, \dots, a_n, b_1, \dots, b_n\}$ ($n \geq 3$) such that

- (1) for all $i \leq n$, $a_i < b_i$,
- (2) for all $i \leq n - 1$, $a_{i+1} < b_i$,
- (3) $a_1 < b_n$, and
- (4) no other relation exists between the elements.

A partial order is *crown-free* if none of its induced suborders are crowns. Kierstead et al. [99] have shown that

- (1) every crown-free recursive partially ordered set with recursive width w has recursive dimension $\leq w!$, and
- (2) For $w \geq 3$, there is a recursive crown-free ordered set with recursive width w , width w , but recursive dimension at least $w \binom{w}{t}$ where $t = \lfloor \frac{1}{2}(w - 1) \rfloor$.

Combining the former result with Theorem 6.24 yields that every crown-free recursive partially ordered set of width w has recursive dimension $\leq (\frac{1}{4}(5^w - 1))!$.

6.7.2 Improving the recursive width

Theorem 6.24 states that a recursive partial order of width w has recursive width $\leq \frac{1}{4}(5^w - 1)$. If further restrictions are made on $\mathcal{P}$, then this can be improved. Kierstead and Trotter [102] showed that if $\mathcal{P}$ is an interval order of width w then it has recursive width $\leq 3w - 2$ (and this covering can be found from the index of $\mathcal{P}$). They also showed that this bound is tight — there are recursive interval orders of width w that have recursive width exactly $3w - 2$. Kierstead et al. [99] showed that if P has width w and recursive dimension d , then the recursive width of P is $\leq \binom{w+1}{2}^{d-1}$.

6.7.3 Height

It is easy to show that if a partial order has height h , then it can be covered by h antichains. Is this true recursively? Schmerl proved (reported in [97]) that it is not, but that a combinatorial modification is true. In particular, he showed that every recursive partial order of height h can be covered by $\binom{h+1}{2}$ recursive antichains, but there are recursive partial orders of height h that cannot be covered by $\binom{h+1}{2} - 1$ recursive antichains. Bounding the recursive dimension does not help: Szemerédi and Trotter showed (reported in [97]) that there exist recursive partial orders of height h and recursive dimension 2 which cannot be covered by $\leq \binom{h+1}{2} - 1$ recursive antichains. The proof we presented for Theorem 6.14 is based on this proof.

Every height- h recursive partial order can be covered by h low antichains. The proof uses the Low Basis Theorem (Theorem 2.12).

7 Miscellaneous results in recursive combinatorics

We state several results in recursive combinatorics without proof.

7.1 Extending partial orders

It is easy to show that any finite partial order $\langle P, \leq \rangle$ has an extension to a linear ordering. In fact, it can even be done efficiently in $O(|P| + |\leq|)$ time [105]. A compactness argument (similar to Theorems 3.3, 4.3, 5.5, and 6.3) shows that this is true for countable partial orders. Perhaps surprisingly, a recursive analogue is *true*, that is, given an index for a recursive partial order one can effectively find an index for a linear extension of it.

Case [34] studied r.e. partial orders. He showed that the r.e. analogue is *false*, that is, there are r.e. orders $\langle P, \leq \rangle$ (both P and the set of ordered pairs $\leq$ are r.e.) that have no r.e. linear extensions. Moreover, he showed that, given any infinite r.e. set A , there is an r.e. partial order $\leq$ on A such that there are no r.e. linear extensions of $\langle A, \leq \rangle$. Roy [144] proved that every recursive partial order has a recursive linear extension and, independent of Case, also proved that there is an r.e. partial order with no r.e. linear extension.

7.2 Vizing's Theorem

An *edge k -coloring* of a graph G is a k -coloring of the edges such that no two incident edges have the same color. The *edge chromatic number* of G , denoted $\eta(G)$, is the least k such that G is edge k -colorable. Recursive edge-colorability and $\eta^r(G)$ are defined in the obvious way.

Vizing [166] ([19] is a more readily available source) showed that, if G has maximal degree d , then $\eta(G) \leq d + 1$. His proof applied only to finite graphs, but by the usual compactness arguments (similar to Theorems 3.3, 4.3, 5.5, and 6.3), it also holds for infinite graphs. There has not been much work done on recursion-theoretic versions of Vizing's Theorem, however Kierstead [95] has shown that if G is a highly recursive graph, then $\eta^r(G) \leq \eta(G) + 1$. This yields a combinatorial modification of Vizing's theorem, namely that if G is highly recursive and has maximum degree d , then $\eta^r(G) \leq d + 2$.

7.3 Graph isomorphism and recursive categoricity

Two graphs are *recursively isomorphic* if there exists a recursive isomorphism between them. A recursive graph G is *recursively categorical* if, for every G' isomorphic to G , G' is actually recursively isomorphic to G . The corresponding notions for highly recursive graphs are defined similarly. Recursive categoricity of models has been extensively studied; see [43].

It is an open problem to determine which (highly) recursive graphs are recursively categorical. Gasarch, Kueker, and Mount [64] have solved the problem for connected highly recursive rooted graphs (i.e., graphs with a distinguished vertex).

Definition 7.1 Let $G = (V, E)$ be a graph such that every vertex has finite degree. An *automorphism* of G is a map $\pi : V \rightarrow V$ that is an isomorphism of G onto itself. $\text{Aut}(G)$ is the set of automorphisms of G . NUMAUT_G is the function that, on input of a nonempty finite function $X \subseteq V \times V$ and a finite sequence of elements of $x_1, \dots, x_n \in V$, outputs the number $|\{(\pi(x_1), \dots, \pi(x_n)) : \pi \in \text{Aut}(G), \pi \text{ extends } X\}|$. Since every vertex of G has finite degree, and X is nonempty, this number is finite.

Gasarch, Kueker, and Mount [64] showed that, if G is a connected highly recursive rooted graph, then G is recursively categorical iff NUMAUT_G is recursive.

7.4 Eulerian and Hamiltonian paths

Definition 7.2 Let $G = (V, E)$ be a graph with $V \subseteq \mathbb{N}$. A *path* in G is a sequence $v_1, v_2, \dots$ such that for every $i \geq 1$, $\{v_i, v_{i+1}\} \in E$. An *Eulerian (Hamiltonian) path* is a path that uses every edge in E (vertex in V) exactly once. A *recursive Eulerian (Hamiltonian) path* is an Eulerian (Hamiltonian) path $v_1, v_2, \dots$ such that there exists a total recursive function f with $f(i) = v_i$. A graph is called *Eulerian (Hamiltonian)* if it has an Eulerian (Hamiltonian) path.⁴

Bean [12] showed that there exist Eulerian (Hamiltonian) recursive graphs with no *recursive* Eulerian (Hamiltonian) paths. For highly recursive graphs the scenario changes dramatically. Bean [12] showed that every Eulerian highly recursive graph *does* have a recursive Eulerian path; moreover, one can effectively find an index for the path given an index for the graph. However, Bean also showed that there are Hamiltonian highly recursive graphs that have no recursive Hamiltonian paths.

Beigel and Gasarch [13], and Harel [76] have studied the complexity of determining if a recursive or highly recursive graph has a (recursive) Eulerian or Hamiltonian path. Beigel and Gasarch showed

- (1) the problem of determining if a recursive graph has a recursive Eulerian (Hamiltonian) path is Σ_3 -complete,
- (2) the same holds for highly recursive graphs, and
- (3) the problem of determining if a recursive graph has an Eulerian path is Π_3 -hard and is in Σ_4 (its exact complexity is not known),
- (4) the problem of determining if a highly recursive graph has an Eulerian path is in Π_2 and is both Σ_1 -hard and Π_1 -hard.

Harel showed that the problem of determining if a (highly) recursive graph has a Hamiltonian path is Σ_1^1 -complete. (This implies that the problem is not in the arithmetic hierarchy.) This is only one of two results in recursive combinatorics whose complexity is outside the arithmetic hierarchy (see Section 7.13 for the other).

⁴The definition of Eulerian (Hamiltonian) graph is nonstandard. Usually the graph is finite and is required to have an Eulerian (Hamiltonian) cycle, i.e., a path that starts at the same vertex where it ends.

The vast difference between determining if a recursive graph has an Eulerian path, and determining if a recursive graph has a Hamiltonian path, might be related to the fact that the Eulerian path problem is in P, while the Hamiltonian path problem is NP-complete (see [61] for a discussion of these concepts). An open problem is to make that analogy rigorous.

7.5 Van Der Waerden's Theorem

Van der Waerden's theorem [167] states⁵ that if $A \subseteq \mathbb{N}$, then either A or $\overline{A}$ has arbitrarily long arithmetic progressions. As an easy corollary, either A or $\overline{A}$ has, for each k , an infinite number of arithmetic progressions of length k . Consider the weaker statement that either A has arbitrarily long arithmetic progressions or $\overline{A}$ has, for each k , an infinite number of arithmetic progressions of length k . Jockusch and Kalantari [89] considered the following 'r.e. version' of the statement: "if A is r.e., then either A has arbitrarily long arithmetic progressions, or there is an r.e. subset of $\overline{A}$ that has, for each k , an infinite number of arithmetic progressions of length k ". They showed that this statement is false, but a finite form of it is true. In particular they showed the following.

- (1) There exists an r.e. set A , such that
 - (a) A has no arithmetic progressions of length 3, and
 - (b) no r.e. subset of $\overline{A}$ has, for each k , an infinite number of arithmetic progressions of length k .
- (2) For every r.e. set A , either
 - (a) A has arbitrarily long arithmetic progressions, or
 - (b) for every k there is an r.e. subset of $\overline{A}$ that has an infinite number of arithmetic progressions of length k .

Gasarch [62] investigated van der Waerden's theorem in a different way. If c is a 2-coloring of $\mathbb{N}$, then a *sequence function* for c is a function that maps k to the ordered pair (a, d) such that there is a k -long monochromatic arithmetic sequence starting at a with difference d . If c is recursive, then there

⁵Our formulation is equivalent to the $c = 2$ case of the standard formulation: for every c and k there exists an n such that if you c -color $\{1, \dots, n\}$ then there is a monochromatic arithmetic progression of length k .

is a recursive sequence function by just looking for an arithmetic sequence until you find one (such will exist by van der Waerden's theorem). He posed the following conjecture:

If $\mathbf{a}$ is a nonrecursive Turing degree, then there exists a coloring $c \in \mathbf{a}$ such that c has no recursive sequence function.

It is easy to show that all weakly 1-generic sets [112] are 2-colorings that satisfy the conjecture, hence the conjecture is true for weakly 1-generic degrees. If the conjecture holds for $\mathbf{a}$, then it holds for all $\mathbf{b}$ such that $\mathbf{a} \leq_T \mathbf{b}$. Hence the conjecture is true for every degree above some weakly 1-generic degree. This includes the 1-generic sets and the n -r.e. sets. These results were proven directly in [62] without using weak 1-genericity (it is easier to use weak 1-genericity).

7.6 Sets of positive density

A set A has *positive upper density* if

$$\lim_{n \rightarrow \infty} \frac{1}{n} |A \cap \{1, \dots, n\}| > 0.$$

It is easy to show that for all sets $A \subseteq \mathbb{N}$, either A or $\overline{A}$ has positive upper density. Consider the following 'r.e. version' of this statement: "if A is r.e., then either A has positive upper density or there is an r.e. subset of $\overline{A}$ that has positive upper density". Jockusch (personal communication) has shown that this statement is false. Let A be a simple set of upper density 0 (which is easily seen to exist by replacing the bound $2e$ by e^2 in Post's simple set construction in [159]). Then, since all r.e. sets disjoint from A are finite, neither A nor any r.e. subset of $\overline{A}$ has positive density.

7.7 Abstract constructions in recursive graph theory

In virtually all the proofs in recursive graph theory, the recursion theory part is 'easy' and the combinatorics is 'hard' or 'clever'. Carstens and Pappinghaus [32] isolated the recursion theory from the combinatorics by proving a general theorem from which, given the proper graph-theoretic constructions, theorems from recursive graph theory can be obtained. They give three examples of theorems that can be obtained in their framework:

- (1) for all $d \geq 3$, there exists a connected highly recursive graph that is d -colorable, but not recursively d -colorable (originally proved in [11]),
- (2) for all $d \geq 2$, there exists a highly recursive d -regular bipartite graph (all vertices have degree d) which has no recursive solution (originally proven in [120]),
- (3) for every $g \geq 1$, there exists a connected highly recursive graph of genus g that cannot be recursively embedded on an orientable surface of genus g (this seems to be new in [32]).

We suspect that the strengthening of (1) that we presented in Theorem 4.30 can be obtained in their framework.

7.8 Relativized results

Carstens [28] considered relativized versions of several of the results stated here. Instead of recursive graphs (bipartite graphs, partitions), he considered $\mathbf{a}$ -recursive graphs, where V and E are recursive in $\mathbf{a}$ ($\mathbf{a}$ -recursive bipartite graphs, etc.). All the negative results relativize easily (e.g., there exists a highly $\mathbf{a}$ -recursive graph which is k -colorable but not $\mathbf{a}$ -recursively k -colorable). For the positive results, he used the relativized version of the Jockusch-Soare low basis theorem (Theorem 2.13).

7.9 Applications to Complexity Theory

Carstens and Pappinghaus [33] use recursive graph theory to show that certain types of algorithms ('extendible algorithms') will not work on several finite problems. The problems considered are matching, maxflow, and integer programming.

7.10 Applications using Σ_1^1 -completeness

David Harel and Tirza Hirst [82] have been working on connecting recursive combinatorics with finite optimization problems. Given an optimization problem A that is based on an NP problem, they have set up a way to define a related problem A^+ in recursive combinatorics. Thus, for example, the infinite version of maximum-clique becomes the question of whether a recursive graph has an infinite clique.

They have shown that if $A^+ \notin \Pi_0^2$ then $A \notin \text{Max-NP}$, and hence $A \notin \text{Max-SNP}$ either (see [132] for definitions). They also have a general result that makes it possible to “lift up” certain NP reductions to become Σ_1^1 reductions. This enables one to prove that, for some A ’s, the recursive counterpart A^+ is Σ_1^1 -complete; hence $A^+ \notin \Pi_0^2$, and so $A \notin \text{Max-NP}$. These two results provide a framework for proving that certain optimization problems are outside Max-NP and Max-SNP.

Arora et al. [7] have shown that, unless $P = NP$, problems that are hard for the class Max-SNP by a certain kind of approximation-preserving reduction, cannot be approximated by a polynomial-time approximation scheme unless $P = NP$ (see their paper for exact definitions). The results of Harel and Hirst show that certain problems are not directly subject to this bad news. Of course, these problems may still be hard to approximate, but the techniques of [7] are probably not able to establish this.

Harel and Hirst [77] have used these two results to prove that many problems in recursive combinatorics are Σ_1^1 -complete. Here is a partial list of the finitary versions, which, as explained, are therefore all outside Max-NP and Max-SNP:

- (1) maximum-clique (this was known to be outside Max-SNP [8]),
- (2) max-independent-set (this is essentially the same as max-clique),
- (3) max-Hamiltonian-path (in the sense that we seek the path having maximum tag, where we tag a path by k if it covers the first k nodes of the graph in some fixed ordering),
- (4) max-set-packing (i.e., the maximal number of nonoverlapping sets from among a given collection of sets),
- (5) complement of min-vertex-cover, this is really the same as max-clique,
- (6) max-subgraph (given graphs H and G , find the subgraph of H with maximal tag that is part of G ; here again we tag a subgraph with k if it covers the first k nodes),
- (7) complement of min-set-cover,
- (8) largest common subsequence (given a set of strings, find the largest string that is a — perhaps noncontinuous — substring of each string in the set),

- (9) max-color (very much like max-independent-set),
- (10) max-exact-cover (even restricted to sets with 3 elements),
- (11) max-domino or max-tiling (maximum k for which the $k \times k$ subgrid of an $n \times m$ grid is tileable).

7.11 Ramsey-Type edge colorings

In this section we do not restrict edge colorings as we did in Section 7.2.

Definition 7.3 A graph is i -connected if removing any $i - 1$ vertices leaves it connected. Let Γ_3 be the set of 3-connected graphs unioned with the triangle graph.

Let $H_1, H_2 \in \Gamma_3$. Burr [24] showed that it is undecidable if a given partial (finite) edge coloring of a highly recursive graph can be extended to a coloring c such that there are no RED H_1 's or BLUE H_2 's. (He actually used a much more restrictive notion than highly recursive.)

Gasarch and Grant [63] showed that there are highly recursive graphs that can be edge colored in a triangle-free manner, but not recursively so colored. They also showed that determining if a particular graph can be recursively colored in a triangle-free manner is Σ_3 -complete.

7.12 Schröder-Bernstein Theorem and Banach's Theorem

The Schröder-Bernstein theorem⁶ states that, if there exists a pair of injections $f : A \rightarrow B$ and $g : B \rightarrow A$, then there exists a bijection $h : A \rightarrow B$. Banach [10] refined this theorem by showing that, if there exist injections $f : A \rightarrow B$ and $g : B \rightarrow A$, then there exist partitions $A = A_1 \cup A_2$ and $B = B_1 \cup B_2$, such that f restricted to A_1 is a bijection between A_1 and B_1 , and g^{-1} restricted to A_2 is a bijection from A_2 to B_2 (in short, $f[A_1] \cup g^{-1}[A_2]$ is a bijection of A onto B).

Remmel [138] showed that the recursive analogue of the Schröder-Bernstein theorem holds, but the recursive analogue of Banach's theorem does not.

⁶Schröder announced the theorem in 1896, but his proof was flawed (see [108] for the full story). Bernstein published the first correct proof in 1898 in [20]. Cantor also had a proof, but it used the axiom of choice, which was not needed.

For both recursive versions, the premise is that there exist partial injections f and g , and recursive sets A and B , such that f (respectively g) is defined on all of A (resp. B). From this, the existence of a partial recursive bijection from A to B (defined on all of A) is easily shown. However, Remmel constructed f, g, A and B such that no recursive A_1, A_2, B_1, B_2 (as in Banach's theorem) exist.

7.13 König's Max-Min Theorem

Let $G = (A, B, E)$ be a finite bipartite graph. A *matching* is a set of disjoint edges. A *cover* is a set of vertices C such that every edge contains a vertex in C . The *matching number* is the maximum cardinality of a matching. The *covering number* is the minimal cardinality of a cover.

König ([107], see [116] for a modern version in English) showed that the matching number and covering number are identical. Lovász and Plummer [116] consider this to be the most important theorem in matching theory. We consider König's matching theorem for countable bipartite graphs. To give the statement substance we need the following special type of cover. A cover C of G is a *König cover* if there exists a matching M such that C can be obtained by picking one vertex from every edge in M .

Aharoni [1, 2] showed that every bipartite graph has a König cover. Aharoni, Magidor, and Shore [3] investigated this theorem in terms of both proof theory and recursion theory. They showed that

- (1) compactness, or König's Lemma on infinite trees, is not enough, from a proof-theoretic viewpoint, to prove the theorem,
- (2) there exist recursive bipartite graphs such that all König covers are of degree above all the hyperarithmetic Turing degrees, and
- (3) for every recursive bipartite graph there exists a König cover of degree $\leq_T \mathcal{O}$ where $\mathcal{O}$ is Kleene's $\mathcal{O}$ (of degree Σ_1^1).

7.14 Arrow's Theorem

Let V be a finite set which we think of as being individuals (or voters). Let X be a finite set which we think of as alternatives being decided upon by the society of individuals (perhaps by voting). Let P be a subset of all rankings of X which we think of as the orders on X that are allowed to be chosen.

Let G be a function that takes the information consisting of every individual's preferred ranking of X (these rankings must be in P) and outputs a ranking in P . The tuple (V, X, P, G) is called a *society*, and is intended to model how a society chooses among alternatives.

Arrow [9] showed that a set of four reasonable conditions on a society imply that there exists a 'dictator', i.e., an individual $v \in V$ such that G will rank X the same way v does. Skala [156] showed that the infinite version of Arrow's theorem depends on the model of set theory. In particular, if ZF is consistent, then there is a model of $\text{ZF} + \text{AC}$ where Arrow's theorem is false for countable V ; however, if one assume the Axiom of Determinacy then there is a model where Arrow's theorem is true for countable V .

Since the classic Arrow's theorem is not true for countable V , recursive combinatorics will play a different role than usual. In this context it is used to recover *some* version of Arrow's theorem that is true. Lewis [114] defined r.e. society, recursive society, and recursive dictator functions. He has shown that an r.e. version of Arrow's theorem, with countable V , is true; and that a recursive version of Arrow's theorem, with countable V , is true with a primitive recursive dictator function.

See [113] and [125, 126, 127] for more on this topic.

7.15 An undecidable problem in finite graph theory

Let G be a finite graph, v be a vertex of G , and $r \in \mathbb{N}$. The r -neighborhood of v is the induced subgraph with vertex set consisting of all vertices of distance at most r from v . Let $r\text{-neib}(G)$ be the set of all r -neighborhoods of a graph G . Note that $r\text{-neib}(G)$ is a set of graphs. Consider the following problem:

Given a finite set of graphs $\{H_1, \dots, H_k\}$, and a number $r \in \mathbb{N}$, does there exist a graph G such that $r\text{-neib}(G) = \{H_1, \dots, H_k\}$?

Winkler [171] has shown that this problem is undecidable in general. However, if the cycle length of G is bounded, then the problem is solvable. This result can be used to solve the following problem: given k and a finite set $D \subseteq \mathbb{N}$, does there exist a k -ary tree whose degree set is D ? This problem had been solved earlier by Winkler [170].

7.16 Hindman's Theorem

Hindman [80] proved the following remarkable theorem: If c is a k -coloring of $\mathbb{N}$, then there exists an infinite monochromatic set X such that every sum of elements from X is the same color. We call such a set *sum-homogeneous*.

Blass, Hirst, and Simpson [18] have analyzed this theorem recursion-theoretically. They have shown

- (1) there exists a recursive 2-coloring of $\mathbb{N}$ such that, for all sum-homogeneous sets X , $X \not\leq_T K$,
- (2) for all k -colorings c of $\mathbb{N}$, there exists a sum-homogeneous set that is recursive in $\emptyset^{\omega+1}$.

7.17 Recursive linear orderings

A *recursive linear ordering* (henceforth RLO) is a linear ordering where the order relation is recursive. For a survey of recursive linear orderings, see [49]. We give one example of a line of research in this area which fits into our theme.

It is a classic theorem that, if L is an infinite linear order, then it has either an infinite ascending or infinite descending suborder. Tennenbaum (see [143]) has shown that this theorem is false recursively, that is, there exist infinite RLO's with no r.e. suborder isomorphic to either ω or ω^* (ω^* is the order $\dots, 3, 2, 1, 0$). Tennenbaum's order is isomorphic to $\omega^* + \omega$. Watnick [169] characterized exactly which order types may have RLO's that are recursive counterexamples. Let $\mathcal{L}$ be the set of all such order types. He showed that $L \in \mathcal{L}$ iff $L \cong \omega + Z\alpha + \omega^*$ where α is a Π_2 linear order (Π_2 base set and Π_2 relation).

7.17.1 Recursive automorphisms

It is easy to see that the order $\omega^* + \omega$ has a non-trivial automorphism. Moses [129] has shown that this is not true recursively. He has shown that an RLO L has a nontrivial automorphism iff L has a dense suborder.

7.18 Well-quasi-orderings

A *quasi-order* $\mathcal{P}$ is a set P (called the *base set*), together with a relation $\leq$ that is transitive and reflexive, but not necessarily anti-symmetric. (e.g.,

take $P = \{0, 1\}^\omega$ and $\leq$ is subsequence). A *well-quasi-order* (henceforth wqo) is a quasi order $\langle P, \leq \rangle$ with the following additional property: if $p_1, p_2, \dots$ is an infinite sequence of elements from P then there exists $i < j$ such that $p_i \leq p_j$.

Kruskal [110] showed that the set of trees, ordered via homeomorphic embedding (or minor), form a wqo (see [130] for an elegant proof). Robertson and Seymour [141] have shown the far more difficult result that the set of all graphs, ordered under minors, is wqo. Another interesting example of a wqo is Σ^* (where Σ is a finite alphabet) under subsequence (proof uses similar techniques to those in [130]).

What makes wqo's interesting is the following theorem: If $\langle P, \leq \rangle$ is a wqo, and $Q \subseteq P$ is closed downward under $\leq$, then there exists a finite number of elements $p_1, \dots, p_k \in P$ such that

$$Q = \{q \in P : \bigwedge_{i=1}^k p_i \not\leq q\}.$$

The set $\{p_1, \dots, p_k\}$ is called the *obstruction set for Q* . For example, since the set of graphs of genus $\leq g$ (some fixed g) is closed under minors, and graphs under minor is a wqo, for every g there is a finite obstruction set $\mathcal{O}_g$ such that G has genus g iff G does not have an element of $\mathcal{O}_g$ as a minor. For genus 1 (planar), the obstruction set is known to be $\{K_5, K_{3,3}\}$. For another example, let Σ be any finite alphabet and let $X \subseteq \Sigma^*$. Let $\text{SUBSEQ}(X)$ be the set of all subsequences of strings in X . Since Σ^* under subsequence is a wqo, and $\text{SUBSEQ}(X)$ is closed under subsequence, there is a finite obstruction set for $\text{SUBSEQ}(X)$. This implies the (somewhat remarkable) theorem that if X is *any* language whatsoever, then the set of subsequences of X is regular. Kruskal [111] notes that this was first proven (using different terminology) by Higman [79] and has been proven several times since then, most recently by Haines [73].

The proof that sets closed downward under $\leq$ have finite obstruction sets is not hard, but it is noneffective. In [16], the recursive analogue is considered and shown to be false.

Harvey Friedman has shown that finite versions of Kruskal's Theorem are unprovable in Peano Arithmetic. See [153] or [115] for a proof, and see [158] for an exposition. Friedman, Robertson, and Seymour have examined proof theoretic considerations of the Graph Minor Theorem [55].

8 Acknowledgments

I thank Richard Beigel, Katia Guimaraes, Tamara Hummel, Henry Kierstead, Georgia Martin, Nick Reingold, and James Schmerl, for proofreading. I thank Tamara Hummel for informing me of the results in her thesis on Ramsey theory. I thank Carl Jockusch for help on Section 3.5. I thank Henry Kierstead for help on Section 6 and offering several helpful suggestions. I thank James Schmerl for several helpful suggestions.

I thank Georgia Martin for finding some subtle mistakes (now corrected, I hope) and in general for proofreading beyond the call of duty.

References

- [1] R. Aharoni, On a duality principle in infinite bipartite graphs, *J. London Math. Soc.*, Ser. 2, **28** (1983) 385–392.
- [2] R. Aharoni, König’s duality theorem for infinite bipartite graphs, *J. London Math. Soc.*, Ser. 2, **29** (1984) 1–12.
- [3] R. Aharoni, M. Magidor and R. A. Shore, On the strength of König’s duality theorem for infinite bipartite graphs, *J. Combin. Theory*, Ser. B, **54** (1992) 257–290.
- [4] K. I. Appel and W. Haken, Every planar map is four colorable. Part I: Discharging, *Illinois J. Math.*, **21** (1977) 429–490.
- [5] K. I. Appel, W. Haken and J. Koch, Every planar map is four colorable. Part II: Reducibility, *Illinois J. Math.*, **21** (1977) 491–567.
- [6] K. I. Appel, W. Haken and J. Koch, Every planar map is four colorable, *Contemp. Math.*, **98** (1989).
- [7] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy, Proof verification and hardness of approximation problems in: *Foundations of Computer Science*, (Proc. 33rd. Annual Sympos. Foundations Comput. Sci., Pittsburg, Oct. 24–27, 1992), FOCS-92, (IEEE Comput. Soc. Press, Los Alamitos, CA, 1992), 14–23.
- [8] S. Arora and S. Safra, Approximating clique is NP-complete, (unpublished manuscript, January 1992).

- [9] K. J. Arrow, Social Choice and Individual Values, Cowles Foundation Res. Econ. Yale Univ., (Cowles Commission Monograph **12**, Wiley, NY and Chapman and Hall, London, 1951; 2nd. edn., 1963; reprinted Yale Univ. Press, New Haven, CT, 1970).
- [10] S. Banach, Un théorème sur les transformations biunivoques, *Fund. Math.*, **6** (1924) 236–239.
- [11] D. R. Bean, Effective coloration, *J. Symbolic Logic*, **41** (1976) 469–480.
- [12] D. R. Bean, Recursive Euler and Hamilton paths, *Proc. Amer. Math. Soc.*, **55** (1976) 385–394.
- [13] R. Beigel and W. I. Gasarch, Σ_3 -complete results in recursive graph theory, (handwritten notes, 1989).
- [14] R. Beigel and W. I. Gasarch, On the complexity of finding the chromatic number of a recursive graph I: The bounded case, *Ann. Pure Appl. Logic*, **45** (1989) 1–38.
- [15] R. Beigel and W. I. Gasarch, On the complexity of finding the chromatic number of a recursive graph II: The unbounded case, *Ann. Pure Appl. Logic*, **45** (1989) 227–246.
- [16] R. Beigel and W. I. Gasarch, On the complexity of finding the obstruction set, (manuscript, 1989).
- [17] R. Beigel and W. I. Gasarch, The mapmaker's dilemma, *Discrete Appl. Math.*, **34** (1991) 37–48, [also in: Univ. Maryland Tech. Rep. TR 2262 (1989)].
- [18] A. R. Blass, J. L. Hirst and S. G. Simpson, Logical analysis of some theorems of combinatorics and topological dynamics, in: *Logic and Combinatorics*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf. Applications Math. Logic to Finite Combinatorics, Humboldt State Univ., Arcata, CA, Aug. 4–10, 1985), S. G. Simpson, (ed.), *Contemp. Math.*, **65** (1987) 125–156.
- [19] B. Bollobás, *Graph Theory: An introductory Course*, Grad. Texts in Math., **63** (1979).

- [20] E. Borel, *Leçons sur la théorie des fonctions*, Gauthier-Villars et fils, Paris, (1898). *First correct proof of Schröder-Bernstein is here.*
- [21] S. Brackin, A summary of Ramsey-type theorems and their provability in weak formal systems, in: *Logic and Combinatorics*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf. Applications Math. Logic to Finite Combinatorics, Humboldt State Univ., Arcata, CA, Aug. 4–10, 1985), S. G. Simpson, (ed.), *Contemp. Math.*, **65** (1987) 169–178.
- [22] R. L. Brooks, On colouring the nodes of a network, *Proc. Cambridge Philos. Soc.*, **37** (1941) 194–197, [also in: *Classic Papers in Combinatorics*, I. Gessel and G.-C. Rota (eds.), (Birkhäuser, Boston, 1987), 118–121].
- [23] N. G. de Bruijn and P. Erdős, A colour problem for infinite graphs and a problem in the theory of relations, *Nederl. Akad. Wentensch. Proc., Ser. A*, **54** (1951) 369–373.
- [24] S. A. Burr, Some undecidable problems involving the edge-coloring and vertex-coloring of graphs, *Discrete Math.*, **50** (1984) 171–177.
- [25] T. J. Carlson and S. G. Simpson, A dual form of Ramsey’s Theorem, *Adv. Math.*, **53** (1984) 265–290.
- [26] T. J. Carlson and S. G. Simpson, Topological Ramsey theory, in: *Mathematics of Ramsey Theory*, J. Nešetřil and V. Rödl, (eds.), *Algorithms Combin.*, **5** (1990) 172–183.
- [27] H.-G. Carstens, *Rekursionstheoretische Untersuchungen über Graphen*, Ph.D. Thesis, Hannover, Germany, (1975). *This is the author’s Habilitationsschrift.*
- [28] H.-G. Carstens, The complexity of some combinatorial constructions, *Z. Math. Logik Grundlag. Math.*, **23** (1977) 121–130.
- [29] H.-G. Carstens, A 5-color extension theorem, in: *Computation Theory and Logic*, E. Börger, (ed.), *Lecture Notes in Comput. Sci.*, **270** (1987) 67–77.
- [30] H.-G. Carstens and U. Golze, Recursive paths in cross-connected trees and an application to cell spaces, *Math. Systems Theory*, **15** (1982) 29–37.
- [31] H.-G. Carstens and P. Päppinghaus, Recursive coloration of countable graphs, *Ann. Pure Appl. Logic*, **25** (1983) 19–45.

- [32] H.-G. Carstens and P. Päppinghaus, Abstract construction of counterexamples in recursive graph theory, in: *Computation and Proof Theory*, (Proc. Logic Colloq., Aachen, July 18–23, 1983, Part II), M. M. Richter, E. Börger, W. Oberschelp, B. Schinzel and W. Thomas, (eds.), *Lecture Notes in Math.*, **1104** (1984) 39–62.
- [33] H.-G. Carstens and P. Päppinghaus, Extensible algorithms, in: *Logic and Machines: Decision Problems and Complexity*, (Proc. Sympos. Rekursive Kombinatorik, May 23–28, 1983, Univ. Münster, Westfalen), E. Börger, G. Hasenjaeger and D. Rödding, (eds.), *Lecture Notes in Comput. Sci.*, **171** (1984) 162–182.
- [34] J. Case, Sortability and extensibility of the graphs of recursively enumerable partial and total orders, *Z. Math. Logik Grundlag. Math.*, **22** (1976) 1–18.
- [35] D. Cenzer and J. B. Remmel, Feasible graphs and colorings, *Math. Logic Quart.*, **41** (1992) 327–352.
- [36] D. Cenzer and J. B. Remmel, Π_1^0 classes in mathematics, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 623–821.
- [37] D. Cenzer and J. B. Remmel, Index sets for Π_1^0 classes, *Ann. Pure Appl. Logic*, (to appear).
- [38] P. A. Cholak, C. G. Jockusch Jr. and T. A. Slaman, Ramsey’s theorem for pairs, *Bulletin of Symbolic Logic*, (1998) 63 (abstract).
- [39] V. Chvátal, *Linear Programming*, (W. H. Freeman, 1983).
- [40] V. Chvátal, Perfectly ordered graphs, in: *Topics on Perfect Graphs*, C. Berge and V. Chvátal, (eds.), *North-Holland Math. Studies*, **88**, *Ann. Discrete Math.*, **21** (1984) 63–65.
- [41] P. Clote, A recursion theoretic analysis of the clopen Ramsey theorem, *J. Symbolic Logic*, **49** (1984) 376–400.
- [42] P. Clote, Weak partition relations, finite games, and independence results in Peano arithmetic, in: *Model Theory of Algebra and Arithmetic*, (Proc. Conf. Appl. Logic, Algebra and Arithmetic, Sept. 1–7, 1979, Karpacz, Poland), L. Packolski, J. Wierzejewski and A. J. Wilkie, (eds.), *Lecture Notes in Math.*, **834** (1984) 92–107.

- [43] J. N. Crossley, A. B. Manaster and M. F. Moses, Recursive categoricity and recursive stability, Logic Paper 49, (Monash Univ., Clayton, Victoria, Australia, 1983).
- [44] G. Dantzig and D. Fulkerson, Minimizing the number of tankers to meet a fixed schedule, Naval Res. Logist. Quarterly, **1** (1954) 217–228.
- [45] G. Dantzig and A. Hoffman, Dilworth's Theorem on partially ordered sets, in: Linear Inequalities and Related Systems, H. W. Kuhn and A. W. Tucker, (eds.), Annals of Math. Stud., **38** (1956) 207–214.
- [46] J. C. E. Dekker, Twilight graphs, J. Symbolic Logic, **46** (1981) 539–571.
- [47] R. P. Dilworth, A decomposition theorem for partially ordered sets, Ann. of Math., Ser. 2, **51** (1950) 161–166.
- [48] R. P. Dilworth, A decomposition theorem for partially ordered sets, in: Classic Papers in Combinatorics, I. Gessel and G.-C. Rota (eds.), (Birkhäuser, Boston, 1987), 139–144.
- [49] R. G. Downey, Recursion theory and linear orderings, Victoria Univ. Res. Rep. 91–71, (June, 1991).
- [50] E. Ellentuck, A new proof that analytic sets are Ramsey, J. Symbolic Logic, **39** (1974) 163–165.
- [51] P. Erdős, Obituary of my friend and coauthor Tibor Gallai, Geombinatorics, **2** (1992) 5–6. *This is more of a newsletter than a journal.*
- [52] P. Erdős and A. Hajnal, On chromatic number of graphs and set-systems, Acta Math. Acad. Sci. Hungar., **17** (1966) 61–99, corr. with B. Rothchild in: On chromatic number of graphs and set-systems, in: (Proc. Cambridge Summer School Math. Logic, Aug. 1–21, 1971), A. R. D. Mathias and H. Rogers, (eds.), Lecture Notes in Math., **337** (1973) 531–538.
- [53] S. Even, A. L. Selman and Y. Yacobi, The complexity of promise problems with applications to public-key cryptography, Inform. and Comput., **61** (1984) 159–173.
- [54] P. C. Fishburn, Interval orders and interval graphs — a study of partially ordered sets, (J. Wiley and Sons, Chichester, 1985).

- [55] H. M. Friedman, N. Robertson and P. Seymour, The metamathematics of the graph minor theorem, in: *Logic and Combinatorics*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf. Applications Math. Logic to Finite Combinatorics, Humboldt State Univ., Arcata, CA, Aug. 4–10, 1985), S. G. Simpson, (ed.), *Contemp. Math.*, **65** (1987) 229–261.
- [56] A. Fröhlich and J. C. Shepherdson, Effective procedures in field theory, *Philos. Trans. Roy. Soc. London, Ser. A*, **248** (1956) 407–432.
- [57] D. R. Fulkerson, Note on Dilworth’s decomposition theorem for partially ordered sets, *Proc. Amer. Math. Soc.*, **7** (1956) 701–702.
- [58] Z. Galil, Efficient algorithms for finding maximum matching in graphs, *Comput. Surveys*, **18** (1986) 23–38.
- [59] T. Gallai and A. N. Milgram, Verallgemeinerung eines graphentheoretischen Satzes von Rédei, *Acta Sci. Math. (Szeged)*, **21** (1960) 181–186.
- [60] F. Galvin and K. Prikry, Borel sets and Ramsey’s theorem, *J. Symbolic Logic*, **38** (1973) 193–198.
- [61] M. R. Garey and D. S. Johnson, *Computers and Intractability, a guide to the theory of NP-completeness*, (W. H. Freeman, San Francisco, 1979).
- [62] W. I. Gasarch, A recursion theoretic view of van der Waerden’s theorem, Technical Report TR-10-84, (Comput. Sci. Dept., Harvard Univ., Cambridge, Mass., 1984).
- [63] W. I. Gasarch and T. Grant, Edge coloring of recursive graphs, (manuscript, 1990).
- [64] W. I. Gasarch, D. W. Keuter and D. M. Mount, Recursive categoricity of highly recursive graphs, in: *Proc. 18th. Manitoba Conf. Numer. Math. and Comput.*, Winnipeg, MB, Sept. 29 – Oct. 1, 1988), D. S. Meek, R. G. Stanton and G. H. J. van Rees, (eds.) *Congr. Numer.*, **69** (1989) 97–102.
- [65] W. I. Gasarch and A. Lee, On the finiteness of the recursive chromatic number, *Ann. Pure Appl. Logic*, (to appear).

- [66] W. I. Gasarch and G. Martin, Index sets in recursive combinatorics, in: Logical Methods, (Papers from Conf. in honor of Anil Nerode's Sixtieth Birthday, June 1–3, 1992, Cornell Univ., Ithaca, NY), J. N. Crossley, J. B. Remmel, R. A. Shore and M. E. Sweedler, (eds.), Progr. Comput. Sci. Appl. Logic, **12** (1993) 352–385.
- [67] K. Gödel, Über formal unentscheidbare Sätze der *Principia mathematica* und verwandter Systeme I, Monatsh. Math. Phys., **38** (1931) 173–198; [translated as: On formally undecidable propositions of *Principia Mathematica* and related systems I, (translated by B. Meltzer, with an introduction by R. B. Braithwaite), (Basic Books, NY, 1963, reprinted Dover, NY, 1992); also in: From Frege to Gödel: A Source Book in Logic, 1879–1931, J. van Heijenoort, (ed.), (Harvard Univ. Press, Cambridge, Mass. and Oxford Univ. Press, London, 1967) 592–617.]
- [68] R. L. Graham, B. L. Rothschild and J. H. Spencer, Ramsey Theory, (Wiley, 1st. edn. 1980, 2nd. edn. 1990).
- [69] J. L. Gross and T. W. Tucker, Topological Graph Theory, (Wiley, 1987).
- [70] A. Gyárfás, On Ramsey Covering-Numbers, in: Infinite and Finite Sets, Vol II, (to P. Erdős on his 60th. birthday, Keszthely, 1973), A. Hajnal, R. Rado and V. T. Sós, (eds.), Coll. Math. Soc. János Bolyai, **10** (1975) 801–816.
- [71] A. Gyárfás, Problems from the world surrounding perfect graphs, (Tanulmányok–MTA Számítástech. Automat. Kutató Int., Budapest, **177**), Zasto. Mat., **19** (1985) 413–441.
- [72] A. Gyárfás and J. Lehel, On-line and first-fit coloring of graphs, J. Graph Theory, **12** (1988) 217–227.
- [73] L. H. Haines, On free monoids partially ordered by embedding, J. Combin. Theory, **6** (1969) 94–98.
- [74] M. Hall, Distinct representatives of subsets, Bull. Amer. Math. Soc., **54** (1948) 922–926.
- [75] P. Hall, On representatives of subsets, J. London Math. Soc., **10** (1935) 26–30, [also in: Classic Papers in Combinatorics, I. Gessel and G.-C. Rota (eds.), (Birkhäuser, Boston, 1987), 58–62].

- [76] D. Harel, Hamiltonian paths in infinite graphs, *Israel J. Math.*, **76** (1991) 317–336.
- [77] D. Harel, (personal communication, 1992).
- [78] P. J. Heawood, Map-colour theorem, *Quarterly J. Math.*, **24** (1890) 332–338.
- [79] A. G. Higman, Ordering by divisibility in abstract algebras, *Proc. London Math. Soc.*, Ser. 3, **2** (1952) 326–336.
- [80] N. Hindman, Finite sums from sequences within cells of partitions on $\mathbb{N}$, *J. Combin. Theory*, Ser. A, **17** (1974) 1–11.
- [81] J. L. Hirst, Marriage theorems and reverse mathematics, in: *Logic and Computation*, (Proc. Workshop, CMU, Pittsburg, June 30 – July 2, 1987), W. Sieg, (ed.), *Contemp. Math.*, **106** (1990) 181–196.
- [82] T. Hirst and D. Harel, Taking it to the limit: on infinite variants of NP-complete problems, in: *Proc. 8th. Annual Structure in Complexity Theory Conf.*, May 18–21, 1993, San Diego, CA, S. Homer, (ed.), *J. Comput. System Sci.*, **53** (1996) 180–193; [another version in: *Structure in Complexity Theory*, (Proc. 8th. Annual IEEE Conf. Struct. Complexity Theory, June 19–23, 1993), (IEEE Comput. Sci. Press, Los Alamitos, CA, 1993), 292–304].
- [83] L. Hopkins, Some Problems Involving Combinatorial Structures Determined by Intersection of Intervals, Ph.D. Thesis, Univ. South Carolina, Columbia, SC, (1981).
- [84] T. L. Hummel, Effective versions of Ramsey’s theorem: avoiding the cone above $\mathbf{0}'$, *J. Symbolic Logic*, **59** (1994) 1301–1325.
- [85] S. Irani, Coloring inductive graphs on-line, in: *Foundations of Computer Science*, (Proc. 31st. Annual Sympos. Foundations Comput. Sci., St. Louis, MO, Oct. 22–24, 1990), FOCS-90, (IEEE Comput. Soc. Press, Los Alamitos, CA, 1990), Vol. 2, 470–479.
- [86] S. Irani, Coloring inductive graphs on-line, *Algorithmica*, **11** (1994) 53–72.
Earlier version in [85].

- [87] C. G. Jockusch Jr., Ramsey's theorem and recursion theory, *J. Symbolic Logic*, **37** (1972) 268–280.
- [88] C. G. Jockusch Jr., (personal communication, 1992).
- [89] C. G. Jockusch Jr. and I. Kalantari, Recursively enumerable sets and van der Waerden's theorem on arithmetic progressions, *Pacific J. Math.*, **115** (1984) 143–153.
- [90] C. G. Jockusch Jr. and R. I. Soare, Π_1^0 classes and degrees of theories, *Trans. Amer. Math. Soc.*, **173** (1972) 33–56.
- [91] C. G. Jockusch Jr. and F. Stephan, A cohesive set which is not high, *Math. Logic Quart.*, **39** (1993) 515–530.
- [92] A. Kanamori and K. McAloon, On Gödel incompleteness and finite combinatorics, *Ann. Pure Appl. Logic*, **33** (1987) 23–41.
- [93] J. Ketonen and R. M. Solovay, Rapidly growing Ramsey functions, *Ann. of Math.*, Ser. 2, **113** (1981) 267–314.
- [94] H. A. Kierstead, An effective version of Dilworth's theorem, *Trans. Amer. Math. Soc.*, **268** (1981) 63–77.
- [95] H. A. Kierstead, Recursive colorings of highly recursive graphs, *Canad. J. Math.*, **33** (1981) 1279–1290.
- [96] H. A. Kierstead, An effective version of Hall's theorem, *Proc. Amer. Math. Soc.*, **88** (1983) 124–128.
- [97] H. A. Kierstead, Recursive ordered sets, in: *Combinatorics and Ordered Sets*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf., Humboldt State Univ., Arcata, CA, Aug. 11–17, 1985), I. Rival, (ed.), *Contemp. Math.*, **57** (1986) 75–102.
- [98] H. A. Kierstead, Recursive and on-line graph coloring, in: *Handbook of Recursive Mathematics*, Yu. L. Ershov, S. S. Goncharov, A. Nerode and J. B. Remmel, (eds.), (Elsevier, Amsterdam, 1998), 1233–1269.
- [99] H. A. Kierstead, G. McNulty and W. T. Trotter Jr., A theory of recursive dimension for ordered sets, *Order*, **1** (1984) 67–82.

- [100] H. A. Kierstead and S. G. Penrice, Radius two trees specify χ -bounded classes, *J. Graph Theory*, **18** (1994) 119–129.
- [101] H. A. Kierstead, S. G. Penrice and W. T. Trotter Jr., On-line coloring and recursive graph theory, *SIAM J. Discrete Math.*, **7** (1994) 72–89.
- [102] H. A. Kierstead and W. T. Trotter Jr., An extremal problem in recursive combinatorics, in: *Combinatorics, Graph Theory and Computing*, Vol. 2, (Proc. 12th. S. E. Conf., Baton Rouge, LA, 1981, Congr. Numer., **33** (1981) 143–153.
- [103] H. A. Kierstead and W. T. Trotter Jr., On-line graph coloring, in: *On-line Algorithms*, (Proc. DIMACS Workshop, Rutgers Univ., New Brunswick, NJ, Feb. 11–13, 1991), L. A. McGeoch and D. D. Sleator, (eds.), DIMACS Series in Discrete Math. Theoret. Comput. Sci., **7** (1992) 85–92.
- [104] S. C. Kleene, *Introduction to Metamathematics*, (D. Van Nostrand, Princeton, 1952, and Elsevier, Amsterdam, 1964, 1971).
- [105] D. E. Knuth, *The Art of Computer Programming*, Vol. 1, *Fundamental Algorithms*, (Addison-Wesley, Reading, MA, 1st. edn. 1969, 2nd. edn. 1973, 3rd. edn. 1997).
- [106] D. König, Sur les correspondances multivoques des ensembles, *Fund. Math.*, **9** (1926) 114–134.
- [107] D. König, *Theorie der endlichen und unendlichen Graphen; kombinatorische Topologie der Streckenkomplexe*, (Akad. Verlag., Leipzig, 1936), [reprinted: Chelsea, 1950, also as: *Theorie der endlichen und unendlichen Graphen*, (with an introduction by P. Erdős and edited by H. Sachs), Teubner-Archiv zur Mathematik, Teubner-Verlag., Leipzig, **6** (1986), also translated as: *Theory of Finite and Infinite Graphs*, translated by R. McCoart and with a commentary by W. T. Tutte, (Birkhäuser, Boston, 1990)].
- [108] A. Korselt, Über einen Beweis des Äquivalenzsatzes, *Math. Ann.*, **70** (1911) 294–296.
- [109] G. Kreisel, Note on arithmetic models for consistent formulæ of the predicate calculus, *Fund. Math.*, **37** (1950) 265–285.

- [110] J. Kruskal, Well-quasi-ordering, the tree theorem and Vazsonyi's Conjecture, *Trans. Amer. Math. Soc.*, **95** (1960) 210–225.
- [111] J. Kruskal, The theory of well-quasi-ordering: a frequently discovered concept, *J. Combin. Theory Ser. A*, **13** (1972) 297–305.
- [112] S. A. Kurtz, Notions of weak genericity, *J. Symbolic Logic*, **48** (1983) 764–770.
- [113] A. A. Lewis, On effectively computable realizations of choice functions, *Math. Social Sciences*, **10** (1985) 43–80.
- [114] A. A. Lewis, An infinite version of Arrow's theorem in the effective setting, *Math. Social Sciences*, **16** (1988) 41–48.
- [115] M. Loebl and J. Matoušek, On undecidability of the weakened Kruskal Theorem, in: *Logic and Combinatorics*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf. Applications Math. Logic to Finite Combinatorics, Humboldt State Univ., Arcata, CA, Aug. 4–10, 1985), S. G. Simpson, (ed.), *Contemp. Math.*, **65** (1987) 275–280.
- [116] L. Lovász and M. D. Plummer, Matching Theory, *Ann. Discrete Math.*, **29** (1986); [also: *North Holland Math. Studies*, **121** (1986); also: *Akadémini Kindó*, Budapest].
- [117] L. Lovász, M. Saks and W. T. Trotter Jr., An on-line graph coloring algorithm with sublinear performance ratio, (Special issue: Graph Theory and Combinatorics, Proc. Cambridge Combinatorial Conf. in honour of P. Erdős, Mar. 21–25, 1988), B. Bollobás, (ed.), *Discrete Math.*, **75** (1989) 319–325.
- [118] K. McAloon, Diagonal methods and strong cuts in models of arithmetic, in: *Logic Colloq. '77*, (Proc. Conf., Warsaw, Aug. 1–12, 1977, dedicated to the memory of A. Mostowski), A. Macintyre, L. Pacholski and J. Paris, (eds.), *Stud. Logic Found. Math.*, **96** (1978) 171–181.
- [119] A. B. Manaster and J. G. Rosenstein, Effective matchmaking (recursion theoretical aspects of a theorem of Philip Hall, *Proc. London Math. Soc.*, Ser. 3, **25** (1972) 615–654.

- [120] A. B. Manaster and J. G. Rosenstein, Effective matchmaking and k -chromatic graphs, *Proc. Amer. Math. Soc.*, **39** (1973) 371–378.
- [121] D. A. Martin, Completeness, the recursion theorem, and effectively simple sets, *Proc. Amer. Math. Soc.*, **17** (1966) 838–842.
- [122] A. R. D. Mathias, On a generalization of Ramsey’s Theorem, *Notices Amer. Math. Soc.*, **15** (1968) 931 (abstract 68T-E19).
- [123] G. Metakides and A. Nerode, Recursively enumerable vector spaces, *Ann. Math. Logic*, **11** (1977) 147–171.
- [124] G. Metakides and A. Nerode, Effective content of field theory, *Ann. Math. Logic*, **17** (1979) 289–320.
- [125] H. R. Mihara, Arrow’s Theorem, Turing Computability, and Oracles, Ph.D. Thesis, Dept. Econ., Univ. Minnesota, MN, (1995).
- [126] H. R. Mihara, Arrow’s theorem and Turing computability, *Econom. Theory*, **10** (1997) 257–276.
- [127] H. R. Mihara, Anonymity and neutrality in Arrow’s theorem with restricted coalition algebras, *Soc. Choice Welf.*, **14** (1997) 503–512.
- [128] D. Misercque, Problème des mariages et récursivité, *Bull. Soc. Math. Belg. Sér. A*, **30** (1978) 111–121.
- [129] M. Moses, Recursive Properties of Isomorphism Types, Ph.D. Thesis, Monash Univ., Clayton, Victoria, Australia, (1983).
- [130] C. St. J. A. Nash-Williams, On well-quasi-ordering finite trees, *Proc. Cambridge Philos. Soc.*, **59** (1963) 833–835; [also in: *Classic Papers in Combinatorics*, I. Gessel and G.-C. Rota (eds.), (Birkhäuser, Boston, 1987), 329–331].
- [131] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, (Prentice-Hall, Englewood Cliffs, New Jersey, 1982).
- [132] C. H. Papadimitriou and M. Yannakakis, Optimization, approximation, and complexity classes, *J. Comput. System Sci.*, **43** (1991) 425–440.

- [133] J. Paris, Some Independence results for Peano arithmetic, *J. Symbolic Logic*, **43** (1978) 725–731.
- [134] J. Paris and L. Harrington, A mathematical incompleteness in Peano arithmetic, in: J. Barwise, (ed.), *Handbook of Mathematical Logic*, *Stud. Logic Found. Math.*, **90** (1977) 1133–1142.
- [135] M. Perles, A proof of Dilworth's decomposition theorem for partially ordered sets, *Israel J. Math.*, **1** (1963) 105–107.
- [136] M. O. Rabin, Computable algebra, general theory and theory of computable fields, *Trans. Amer. Math. Soc.*, **95** (1960) 341–360.
- [137] F. Ramsey, On a problem of formal logic, *Proc. London Math. Soc.*, Ser. 2, **30** (1930) 264–286; [also in: *Classic Papers in Combinatorics*, I. Gessel and G.-C. Rota (eds.), (Birkhäuser, Boston, 1987), 2–24].
- [138] J. B. Remmel, On the effectiveness of the Schröder-Bernstein theorem, *Proc. Amer. Math. Soc.*, **83** (1981) 379–386.
- [139] J. B. Remmel, Graph colorings and recursively bounded Π_1^0 -classes, *Ann. Pure Appl. Logic*, **32** (1986) 185–194.
- [140] G. Ringel and J. W. T. Youngs, Solution of the Heawood map-coloring problem, *Proc. Nat. Acad. Sci. U.S.A.*, **60** (1968) 438–445.
- [141] N. Robertson and P. D. Seymour, Graph minors XV: Wagner's conjecture, *J. Combin. Theory*, Ser. B, (to appear).
- [142] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, (1st. edn., McGraw-Hill, New York-Toronto, Ont.-London, 1967; 2nd. edn., MIT Press, Cambridge, Mass., London, 1987).
- [143] J. G. Rosenstein, Linear orderings, *Pure Appl. Math.*, **98** (1982).
- [144] D. K. Roy, Effective extensions of partial orders, *Z. Math. Logik Grundlag. Math.*, **36** (1990) 233–236.
- [145] J. H. Schmerl, Recursive colorings of graphs, *Canad. J. Math.*, **32** (1980) 821–830.

- [146] J. H. Schmerl, The effective version of Brooks' theorem, *Canad. J. Math.*, **34** (1982) 1036–1046.
- [147] J. H. Schmerl, (personal communication, 1992)
- [148] D. Seetapun and T. A. Slaman, On the strength of Ramsey's Theorem, *Notre Dame J. Formal Logic*, **36** (1995) 570–582.
- [149] J. R. Shoenfield, Degrees of models, *J. Symbolic Logic*, **25** (1960) 233–237.
- [150] J. Silver, Every analytic set is Ramsey, *J. Symbolic Logic*, **35** (1970) 60–64.
- [151] S. G. Simpson, Sets which do not have subsets of every higher degree, *J. Symbolic Logic*, **43** (1978) 135–138.
- [152] S. G. Simpson, Which set existence axioms are needed to prove the Cauchy/Peano theorem for ordinary differential equations?, *J. Symbolic Logic*, **49** (1984) 783–802.
- [153] S. G. Simpson, Nichtbeweisbarkeit von gewissen kombinatorischen Eigenschaften endlicher Baume, *Arch. Math. Logik Grundlag.*, **25** (1985) 45–65; [translated as: Nonprovability of certain combinatorial properties of finite trees, in: Harvey Friedman's Research on the Found. of Math., L. A. Harrington, M. D. Morley, A. Šcedrov and S. C. Simpson, (eds.), *Stud. Logic Found. Math.*, **117** (1985) 87–117].
- [154] S. G. Simpson, Recursion theoretic aspects of the dual Ramsey theorem, in: *Recursion Theory Week*, (Proc. Conf., Oberwolfach, Apr. 15–21, 1984), H.-D. Ebbinghaus, G. H. Müller and G. E. Sachs, (eds.), *Lecture Notes in Math.*, **1141** (1985) 357–371.
- [155] S. G. Simpson, Unprovable theorems and fast-growing functions, in: *Logic and Combinatorics*, (Proc. AMS-IMS-SIAM Joint Summer Research Conf. Applications Math. Logic to Finite Combinatorics, Humboldt State Univ., Arcata, CA, Aug. 4–10, 1985), S. G. Simpson, (ed.), *Contemp. Math.*, **65** (1987) 359–394.
- [156] H. J. Skala, On the foundations of the social ordering problem, in: *Game Theory and Mathematical Economics*, D. Moeschlin and D. Pallaschke, (eds.), (North Holland, 1981), 249–261.

- [157] T. A. Slaman, personal communication, communicated by Carl Jockusch, 1992.
- [158] C. Smoryński, The varieties of arboreal experience, *Math. Intelligencer*, **4** (1982) 182–189.
- [159] R. I. Soare, Recursively enumerable sets and degrees, *Bull. Amer. Math. Soc. (N.S.)*, **84** (1978) 1149–1181.
- [160] R. M. Solovay, Hyperarithmetically encodable sets, *Trans. Amer. Math. Soc.*, **239** (1978) 99–122.
- [161] E. Specker, Ramsey's Theorem does not hold in recursive set theory, in: *Logic Colloq. '69*, (1971) 439–442.
- [162] D. P. Sumner, Subtrees of a graph and the chromatic number, in: *The Theory and Applications of Graphs*, G. Chartrand, (ed.) (John Wiley and Sons, NY, 1981), 557–576.
- [163] K. Tanaka, A game-theoretic proofs [proof] of analytic Ramsey theorem *Z. Math. Logik Grundlag. Math.*, **38** (1992) 301–304.
- [164] H. Tverberg, On Schmerl's effective version of Brooks's theorem, *J. Combin. Theory, Ser. B*, **37** (1984) 27–30.
- [165] S. Vishwanathan, Randomized online graph coloring, *J. Algorithms*, **13** (1992) 657–669.
- [166] V. G. Vizing, The chromatic class of a multigraph, *Cybernetics*, 1, No. 3 (1965) 32–41. *Appeared originally in Russian in Kibernetika (Kiev) 1965.*
- [167] B. L. van der Waerden, Beweis einer Baudetschen vermutung, *Nieuw Arch. Wisk.*, **15** (1927) 212–216.
- [168] H. Wang, *Popular Lectures on Mathematical Logic*, (Van Nostrand Reinhold Co., New York-London; and Kexue Chubanshe (Science Press), Beijing, 1981).
- [169] R. Watnick, A generalization of Tennenbaum's Theorem on effectively finite recursive linear orderings, *J. Symbolic Logic*, **49** (1984) 563–569.

- [170] P. Winkler, Graphic characterization of k -trees, *Congr. Numer.*, **33** (1981) 349–357.
- [171] P. Winkler, Existence of graphs with a given set of r -neighborhoods, *J. Combin. Theory, Ser. B*, **34** (1983) 165–176.

William Gasarch
Dept. of Computer Science and
Inst. for Advanced Computer Studies
University of Maryland
College Park, MD 20742, USA
email: gasarch@cs.umd.edu

Chapter 17

Constructive Abelian Groups

N. G. Khisamiev

Contents

Introduction

1. Operations over constructive groups
2. Constructivizations with a recursively enumerable basis
3. Direct sums of cyclic and quasicyclic p -groups
4. Constructive abelian p -groups
5. Strong constructivity and Kulikov's basis
6. Torsion-free abelian groups
7. Model theoretic method in the theory of constructivizable abelian groups
8. Connections between constructivizability and strong constructivizability
9. Constructivizability of subgroups and factor groups
10. The arithmetic hierarchy of abelian groups

References

Introduction

In the theory of constructivizable models, the basic problems are those of existence, uniqueness and extensions of constructivizations. Investigations of these problems are developed in the context of general model theory for specific classes of algebraic systems. The present chapter is devoted to an exposition of basic results of the theory of constructivizable abelian groups. Investigations in this field began with the work of Mal'tsev [67]. Further results in this field are contained in the works of S. S. Goncharov, V. V. Dzgoev, V. P. Dobritsa, N. G. Khisamiev, C. Lin, A. B. Molokov, A. T. Nurtazin, F. Richman, R. Smith and many others. The main sources for the theory of constructivizable models are the survey of Mal'tsev [66] and the books of Ershov [19] and Goncharov [31].

Algorithmic problems in group theory (the word problem, the conjugacy problem, etc.) arose before the appearance of a precise concept of an algorithm. At the beginning of the 1930's, this concept was made more precise, and the first results of the theory of algorithms were obtained. On the basis of these results, Novikov [75] proved the undecidability of the word problem for groups. At first, algorithmic problems were considered only for finitely presented groups. Later, the importance of studying recursively enumerably presented groups was clarified. The concept of a numerable group was introduced in order to study recursively enumerably presented groups in connection with the rich theory of algorithms. A numerable group is called *constructivizable* if there is an algorithm which, for any numbers m, n, s of elements a, b, c of the group, answers the question: is it true that $ab = c$? We give some notation and definitions. Let ω denote the set of all natural numbers, P the set of all prime numbers, p a prime number, $(x_0, \dots, x_{n-1})$ an ordered sequence with $x_i \in \omega$, $i \leq n-1$, and $\langle x_0, \dots, x_{n-1} \rangle$ the number of $(x_0, \dots, x_{n-1})$. Let $\mathbb{Z}_{p^n}$ be the cyclic group of order p^n and $\mathbb{Z}_{p^\infty}$ be the quasi cyclic p -group. The sign $\oplus$ denotes the direct sum. By the term *group* we mean a countable abelian group. Let G be a group and $M \subseteq G$ be some subset. We use $\text{gr}(M)$ to denote the subgroup of G generated by the set M . If $M = \{g_1, \dots, g_n\}$ is finite, then we write $(g_1, \dots, g_n)$ instead of $\text{gr}(g_1, \dots, g_n)$. The order of an element $g \in G$ is denoted by $|g|$. If α is an ordinal number, then G^α denotes the direct sum of α copies of G .

We now define a (strongly) constructivizable group. Fix a numeration γ of the set of all formulae $A(v_0, \dots, v_{s-1})$, $s \in \omega$, with free variables $v_0, \dots, v_{s-1}$ in the signature σ consisting of the binary operation $+$. A map ν of the set ω onto a group G is called a numeration of this group. The pair (G, ν) is called a numerable group. We take

$$D_\nu(G) = \{\langle m, n, s \rangle \mid G \models (\nu n + \nu m = \nu s)\}$$

$$D_\nu^*(G) = \{\langle m, n_0, \dots, n_{s-1} \rangle \mid G \models \gamma m(\nu n_0, \dots, \nu n_{s-1})\}$$

Let X be a subset of ω . The pair (G, ν) is called (*strongly*) X -constructive if the set $(D_\nu^*(G)) \cap D_\nu(G)$ is X -recursive. If X denotes some class of subsets of ω , then we replace the phrase "is X -recursive" by "belongs to the class X ". A group G is called (*strongly*) X -constructivizable if there exists a numeration ν of G such that the pair (G, ν) is (*strongly*) X -constructive. If X is a recursive set, then instead of " X -constructivizable" and " X -constructivizable"

we write “constructive” and “constructivizable”. The proofs of all assertions which contain X as an oracle in the formulations are given only for $X = \emptyset$.

1 Operations over constructive groups

The simplest examples of groups are the cyclic group $\mathbb{Z}_{p^n}$ of order p^n , the quasicyclic group $\mathbb{Z}_{p^\infty}$, the additive group $\mathbb{Z}$ of integer numbers, and the additive group $\mathbb{Q}$ of rational numbers. The elements of $\mathbb{Z}_{p^n}$ are $0, 1, \dots, p^n - 1$, and the group operation is addition modulo p^n . Define the numeration $\alpha : \omega \rightarrow \mathbb{Z}_{p^n}$ by the rule: if $x = p^n m + k$, $0 \leq k < p^n$, then $\alpha_n x = k$. It is easy to check that $(\mathbb{Z}_{p^n}, \alpha_n)$ is a constructive group.

The quasicyclic group $\mathbb{Z}_{p^\infty}$ is the set of all complex roots of all equations of the form $x^{p^n} = 1$, $n \in \omega$ with the usual multiplication. Let ξ_n be a primitive root of the equation $x^{p^n} = 1$. $\mathbb{Z}_{p^\infty}$ is generated by the elements ξ_n , $n \in \omega$. Define the numeration β_p of this group by the rule: if $x = p^n + k$, $0 \leq k < p^n$, then $\beta_p x = k \xi_n$. It is easy to verify that $(\mathbb{Z}_{p^\infty}, \beta_p)$ is a constructive group. We can define constructive numerations γ_1 and δ_1 of the groups $\mathbb{Z}$ and $\mathbb{Q}$ similarly. Let the groups $\mathbb{Z}_{p^n}$, $\mathbb{Z}_{p^\infty}$, $\mathbb{Z}$ and $\mathbb{Q}$ have the numerations α_n , β_p , γ_1 and δ_1 respectively.

We next show that these groups are in fact strongly constructive. Define the one-place predicates $p^s |$, $p \in P$, $s \in \omega$ by the rule $\exists y(p^s y = x)$. Say that the element x is divisible by p^s in the group A if and only if the predicate $p^s |$ is true of x . Let $\sigma' = \langle +, p^s | ; p \in P, s \in \omega \rangle$. We say that a numerable group (A, ν) is X -constructive with an algorithm for divisibility, if (A, ν) is X -constructive in the signature σ' .

Proposition 1.1 *A numerable abelian group (A, ν) is strongly X -constructive if and only if the theory, $\text{Th}(A)$, of A is X -recursive and the pair (A, ν) is X -constructive with an algorithm for divisibility.*

Indeed, Ershov [19] proved that the complete theory of abelian groups in the signature σ' is model-complete. By Proposition 5 from Ershov [19, p. 316], it follows that the pair (A, ν) is strongly constructive. $\square$

Corollary 1.2 *The groups $\mathbb{Z}_{p^n}$, $\mathbb{Z}_{p^\infty}$, $\mathbb{Z}$ and $\mathbb{Q}$ are strongly constructivizable.*

Indeed, it is easy to compute the Szmielew invariants (see §7) of these groups. Therefore the theories of these groups are decidable. $\square$

We introduce two important operations over constructive groups. A sequence of (strongly) X -constructive groups $\{(G_i, \mu_i) \mid i \in \omega\}$ is called X -computable if the sequence of sets $(\{D_{\mu_i}^*(G_i)\} \{D_{\mu_i}(G_i)\})$ is X -computable. Fix a Gödel numbering of all finite sequences of natural numbers. This means that for any given sequence of numbers $(x_0, \dots, x_{n-1})$ we can effectively compute its Gödel number $\langle x_0, \dots, x_{n-1} \rangle$, and vice versa. Let $G = \bigoplus \{G_i \mid i \in \omega\}$ be the direct sum of groups G_i . The numeration μ of G is defined by

$$\mu x = (\mu_0 x_0, \dots, \mu_{n-1} x_{n-1})$$

where $x = \langle x_0, \dots, x_{n-1} \rangle$. The pair (G, μ) is called the *direct sum* of the numerable groups $\{(G_i, \mu_i)\}$, and is denoted by $(G, \mu) = \bigoplus \{(G_i, \mu_i) \mid i \in \omega\}$.

Proposition 1.3 *The direct sum of an X -computable sequence*

$$\{(G_i, \mu_i) \mid i \in \omega\}$$

of (strongly) X -constructive groups is (strongly) X -constructive.

Hence the numerable groups $(\mathbb{Q}^\omega, \delta) = (\mathbb{Q}, \delta_1)^\omega$, $(\mathbb{F}_\omega, \gamma) = (\mathbb{Z}, \gamma_1)^\omega$, and $(\mathbb{U}, \beta) = \bigoplus \{(\mathbb{Z}_{p^\infty}, \beta_p)^\omega \mid p \in P\} \oplus (\mathbb{Q}^\omega, \delta)$ are strongly constructive. Here $\mathbb{F}_\omega$ is a free abelian group of countable rank. Consider the groups $\mathbb{U}$, $\mathbb{F}_\omega$ and $\mathbb{Q}^\omega$ as the numerable groups $(\mathbb{U}, \beta)$, $(\mathbb{F}_\omega, \gamma)$ and $(\mathbb{Q}^\omega, \delta)$ respectively.

We can show that the condition of computability of the sequence $\{(G_i, \mu_i)\}$ is essential. Indeed, if the group A is constructive, the set of prime numbers $\{p \mid A \models \exists x(px = 0)\}$ is recursively enumerable. Let S be some non-recursively enumerable set of prime numbers. The group $\bigoplus \{\mathbb{Z}_p \mid p \in S\}$ is not constructivizable, although each of the summands $\mathbb{Z}_p$ is constructive.

Let (A, ν) be a numerable set. A subset $A_0 \subseteq A$ is called *recursively enumerable* if there is a recursively enumerable set S_0 of the natural numbers such that $\nu S_0 = A_0$. Let (A, ν) , (B, μ) be numerable sets, and Φ be a map from A into B . If there exists a recursive function φ such that $\Phi \nu n = \mu \varphi(n)$ for any number $n \in \omega$, then Φ is called a *recursive map* of the numerable set (A, ν) into the numerable set (B, μ) . The notions of a recursive homomorphism, a recursive isomorphism, etc., are defined analogously.

Proposition 1.4 *Let A_0 be an X -recursively enumerable subgroup of an X -constructive group (A, ν) . There is an X -constructivization ν_0 of the subgroup A_0 such that the embedding of A_0 into A by the identity is an X -recursive isomorphism of the numerable group (A_0, ν_0) into (A, ν) .*

From here it follows that any recursively enumerable subgroup of the group $\mathbb{U}$ defined above is constructivizable. In the following section, we will prove that the converse is also true; i.e., any constructivizable abelian group is isomorphic to some recursively enumerable subgroup of $\mathbb{U}$.

Corollary 1.5 *The torsion part of an X -constructivizable group is X -constructivizable.*

Let A be a torsion abelian group. The subgroup

$$A_p = \{a \in A \mid \exists n(p^n x = 0)\}$$

of A is called the p -component of A . It is known that $A = \bigoplus \{A_p \mid p \in P\}$.

Corollary 1.6 *Let (A, ν) be a (strongly) X -constructive torsion abelian group. There exist numerations ν_p of the p -components A_p , $p \in P$, of A such that the sequence $\{A_p, \nu_p\}$ is an X -computable sequence of (strongly) X -constructive groups, and the pair (A, ν) is X -recursively isomorphic to the direct sum $\bigoplus \{(A_p, \nu_p) \mid p \in P\}$ of the numerable groups (A_p, ν_p) .*

Proof. Let (A, ν) be a (strongly) constructive group. The subgroups A_p , $p \in P$, are recursively enumerable. Define the numerations ν_p of subgroups A_p as in Proposition 1.4. If (A, ν) is a strongly constructive group, then from Proposition 1.1 and the decidability of $\text{Th}(A_p)$, it follows that the groups (A_p, ν_p) are strongly constructive. For every element $\nu x \in A$ it is possible to find numbers $x_0, \dots, x_{n-1}$ such that $\nu x_i \in A_{p_i}$, $i \leq n-1$, and

$$\nu x = \nu x_0 + \dots + \nu x_{n-1}.$$

Indeed, let $x \in \omega$, and suppose the order of νx is $m > 0$ with $m = p_0^{k_0} \dots p_{n-1}^{k_{n-1}}$ and $m_i = m/p_i^{k_i}$, $i \leq n-1$. There are numbers $u_i \in \mathbb{Z}$ such that

$$1 = u_0 m_0 + \dots + u_{n-1} m_{n-1}.$$

Hence $\nu x = u_0 m_0 \nu x + \dots + u_{n-1} m_{n-1} \nu x$. The elements $\nu x_i = u_i m_i \nu x$ are the ones for which we were looking. $\square$

From Corollary 1.6 it follows that the problem of the existence of a constructivization in the class of torsion abelian groups reduces to the same problem for the class of abelian p -groups.

2 Constructivizations with a recursively enumerable basis

In this section we show that the existence problem of constructivizations for the class of abelian groups can be reduced to the same problem for the classes of periodic groups and torsion-free groups. Theorems for existence of a (strong) constructivization with a recursively enumerable basis and some of their corollaries are given.

2.1 Constructive extensions

Let (B, ν) and (C, μ) be constructive groups. Denote the elements of B by $a, b, \dots$, and the elements of C by $u, v, \dots$. A *system of factors* from C into B is a function $\Phi : C \times C \rightarrow B$ which satisfies the following conditions for all $u, v, w \in C$.

- (i) $\Phi(u, v) = \Phi(v, u)$.
- (ii) $\Phi(u, v) + \Phi(u + v, w) = \Phi(u, v + w) + \Phi(v, w)$.
- (iii) $\Phi(u, 0) + \Phi(0, v) = 0$.

The system of factors Φ is called *recursive* if there is a recursive function $\varphi : \omega^2 \rightarrow \omega$ such that $\Phi(\mu n, \mu m) = \nu \varphi(n, m)$. Given the groups B, C and a system of factors Φ from C into B , we define a group A as follows: the elements of A are pairs (u, a) , $u \in C, a \in B$ and the group operation is given by

$$(u, a) + (v, b) = (u + v, a + b + \Phi(u, v))$$

A is called an *extension* of B by C respecting the system of factors Φ . We define a numeration γ of A by $\gamma n = (\mu r, \nu s)$, where $n = \langle r, s \rangle$. It is easy to verify that if the system of factors is recursive, then the group (A, γ) is constructive. The pair (A, γ) is called an extension of (B, ν) by (C, μ) respecting the system of factors Φ .

Proposition 2.1 *Let (A, γ) be an X -constructive group and B be a subgroup of A . The following conditions are equivalent.*

- (i) *The subgroup B is X -recursive in (A, γ) .*

- (ii) *There are X -constructivizations ν and μ of B and A/B respectively, and an X -recursive system of factors Φ from $(A/B, \mu)$ into (B, ν) such that the pair (A, γ) is recursively isomorphic to the extension of (B, ν) by $(A/B, \mu)$ respecting the system Φ .*

Proof. From the definition of an extension of a numerable group it follows that (ii) implies (i). We prove that (i) implies (ii). From the recursiveness of B it follows that there is a recursively enumerable set of representative $\{u_i \mid i \in \omega\}$ of cosets of A/B . We can suppose that $u_0 = 0$. Derive both a numeration μ of A/B and a system of factors Φ by setting $\mu i = \bar{u}_i$ and $\Phi(\bar{u}_i, \bar{u}_j) = u_k - u_i - u_j$, where $\bar{u}_k = \bar{u}_i + \bar{u}_j$. $\square$

Let (A, ν) be a constructive group and T be its periodic part. It is clear that T is recursively enumerable. As a matter of fact, there is a constructivization μ of A such that T is a recursive subgroup of (A, μ) .

A set of elements $a_0, \dots, a_{n-1}$ of A is called *linearly independent* if and only if the equality

$$\alpha_0 a_0 + \dots + \alpha_{n-1} a_{n-1} = 0,$$

$\alpha_i \in \mathbb{Z}, i \leq n-1$, implies $\alpha_i a_i = 0$ for all i . A maximal linearly independent set of elements of A , each of which has either prime order or infinite order, is called a *basis* of A . The cardinality of a basis of A is called the *rank* of A .

Theorem 2.2 (Dobritsa [7], Nurtazin [76]). *Every X -constructivizable abelian group A has an X -constructivization μ possessing the following property: there is an X -recursively enumerable set of elements $\{c_i \mid i \in I\}$ such that the cosets $\{c_i + T\}$ form a basis of the factor group A/T of A by its periodic part T .*

Proof. We can suppose that the basis $\{c_i + T\}$ of A/T is infinite. Let ν be a constructive numeration of A . The elements $a_0, \dots, a_{k-1}$ of A are called *t -dependent* if there are numbers $m_i, i \leq k-1, |m_i| \leq t$ and not all m_i equal to zero, such that $m_0 a_0 + \dots + m_{k-1} a_{k-1} = 0$. We construct the required numeration μ in stages. Suppose t steps have been completed, and a set of *t -independent* elements $C^t = \{c_0^t, \dots, c_i^t\}$ has been defined. The set

$$C_t = \{y \in A \mid \alpha y = \alpha_0 c_0^t + \dots + \alpha_t c_t^t, |\alpha|, |\alpha_i| \leq t, \alpha \neq 0\}$$

is called the *t -closure* of the set C^t .

STEP $t + 1$: We set $c_{t+1}^t = 0$. Let $i \leq t + 1$ be the least number such that c_i^t is $(t + 1)$ -dependent on $c_0^t, \dots, c_{i-1}^t$. Find the elements c'_{i+j} , $j \leq t + 1 - i$ with the least ν -numbers such that the set

$$\{c_0^t, \dots, c_{i-1}^t, c'_i, \dots, c'_{t+1}\}$$

is $(t + 1)$ -independent. Set $c_k^{t+1} = c_k^t$, $k < i$, $c_{i+j}^{t+1} = c'_{i+j} + (t + 1)! c_{i+j}^t$, $j \leq t + 1 - i$. It is easy to verify that the map $\varphi(c_s^t) = c_s^{t+1}$, $s \leq t$, is an isomorphic embedding of the partial group C_t (relative to the predicate $x + y = z$) into the partial group C_{t+1} . We can define a numeration μ^{t+1} of the set C_{t+1} . Suppose the numeration $\mu^t : N^t \rightarrow C_t$ has been defined, where $N^t = \{0, 1, \dots, s^t\}$, and

$$C_{t+1} \setminus C_t = \{a_0, \dots, a_{m-1}\}.$$

We take $\mu^{t+1}x = \varphi\mu^tx$ for $x \leq s^t$ and $\mu^{t+1}(s^t + i) = a_{i-1}$, $1 \leq i \leq m$. Step $t + 1$ is finished. $\square$

Corollary 2.3 *Every X -constructivizable abelian group has an X -constructive numeration μ such that the periodic part T is X -recursive in (A, μ) .*

Indeed, let the numeration μ and the set $\{c_i \mid i \in I\}$ be as in Theorem 2.2. For every element μx we can effectively find numbers $s, m_0, \dots, m_{n-1}$ such that

$$\mu x = \mu s + m_0 c_0 + \dots + m_{n-1} c_{n-1}.$$

Thus $\mu x \in T$ if and only if $m_0 = \dots = m_{n-1} = 0$. Hence the subgroup T is recursive in (A, μ) . $\square$

We have already defined the notion of X -constructive with an algorithm for divisibility. In a similar way, we can define X -constructive with an algorithm for linear independence and X -constructive with an algorithm for the order of elements. Let the numeration μ of A and the set $\{c_i \mid i \in I\}$ be defined as in Theorem 2.2. It is clear that we can choose a recursively enumerable basis $\{t_j \mid j \in J\}$ of the periodic part T of A . Thus the system

$$\{t_j \mid j \in J\} \cup \{c_i \mid i \in I\} \quad (*)$$

is the basis of A . From here we obtain:

Corollary 2.4 *Every X -constructivizable abelian group has an X -constructivization with an algorithm for the order of its elements and for linear independence.*

The strongly constructive group

$$\mathbb{U} = \bigoplus \{\mathbb{Z}_{p^\infty}^\omega \mid p \in P\} \oplus \mathbb{Q}^\omega$$

was defined in §1. It is evident that every recursive map of the basis $(*)$ into $\mathbb{U}$ can be extended to a recursive embedding of (A, μ) into $\mathbb{U}$. Therefore the following corollary holds.

Corollary 2.5 *An arbitrary abelian group is X -constructivizable if and only if it is isomorphic to some X -recursively enumerable subgroup of $\mathbb{U}$.*

From Proposition 2.1 and Corollary 2.3 we obtain:

Corollary 2.6 *An arbitrary abelian group A is X -constructivizable if and only if there are X -constructivizations ν and μ of its periodic part T and the factor group A/T respectively, such that the group A is an extension of (T, ν) by $(A/T, \mu)$ respecting some X -recursive system of factors.*

Hence the existence problem of constructivizations for the class of abelian groups is reduced to the same problem for the classes of periodic groups and torsion-free groups.

Remark 2.1 If the factor group A/B is finite, then any system of factors from A/B into B is recursive. Hence any extension of a constructivizable group by a finite group is constructivizable.

P. Smith [87] showed that Corollary 2.5 can be made stronger. A group D is called divisible if for every integer $n > 0$ and for every $g \in D$, the equation $nx = g$ has at least one solution in D . The pair (D, φ) is called a divisible closure of A if D is a divisible group, $\varphi : A \rightarrow D$ is an embedding, and for any $d \in D$, $d \neq 0$, there are $m \in \omega$, and $g \in A$, $g \neq 0$, such that $md = \varphi(g)$. A constructive group (D, α) is called a divisible closure of a constructive group (A, ν) if there is a recursive embedding $\varphi : (A, \nu) \rightarrow (D, \alpha)$ and (D, φ) is a divisible closure of A .

Theorem 2.7 (P. Smith [87]). *Every X -constructive abelian group has an X -constructive divisible closure.*

2.2 A generalization

We next consider a generalization of Theorem 2.2 and some of its corollaries.

Theorem 2.8 (Khisamiev [48]). *Let (A, ν) be a (strongly) X -constructive group, and B be an X -recursively enumerable subgroup such that the factor group A/B is torsion-free. There is a numeration μ of A for which the following properties are true.*

- (i) *The group (A, μ) is (strongly) X -constructive.*
- (ii) *The subgroup B is X -recursive in (A, μ) .*
- (iii) *There is an X -recursively enumerable set $\{c_i \mid i \in I\}$ of elements of (A, μ) such that the cosets $\{c_i + B\}$ form a basis of A/B .*

Remark 2.2 Theorem 2.8 remains true when “strongly X -constructive” is replaced by “ X -constructive with an algorithm for divisibility”.

Corollary 2.9 *Let (A, ν) be a (strongly) X -constructive abelian group, and T be its periodic part. There is a (strong) X -constructivization μ of A such that the subgroup T is X -recursive in (A, μ) .*

Corollary 2.10 *Let (A, ν) be a (strongly) X -constructive torsion-free abelian group, and B be an X -recursively enumerable pure subgroup. There is a (strong) X -constructivization μ of A such that the subgroup B is X -recursive in (A, μ) .*

Corollary 2.11 *If (A, ν) is an X -constructive torsion-free abelian group, and B is an X -recursively enumerable pure subgroup, then the factor group A/B is X -constructivizable.*

Corollary 2.12 *Any X -recursively enumerable definable torsion-free abelian group is X -constructivizable.*

Proof. Let $\mathbb{F}$ be a free abelian group of countable rank. It was shown in §1 that $\mathbb{F}$ is strongly constructive. We can choose a recursively enumerable subgroup B of $\mathbb{F}$ such that the factor group $\mathbb{F}/B$ is isomorphic to A . As the group A is torsion-free, it follows that B is pure in $\mathbb{F}$. Hence by Corollary 2.11, A is constructivizable. $\square$

Latkin [61] announced that Corollary 2.12 is not true for 2-step nilpotent torsion-free groups.

The strongly constructive group $\mathbb{Q}^\omega$ was defined in §1.

Corollary 2.13 *A torsion-free abelian group A is X -constructivizable with an algorithm for divisibility if and only if it is isomorphic to some X -recursive subgroup A' of $\mathbb{Q}^\omega$.*

Theorem 2.7 is also true for strongly constructivizable groups.

Theorem 2.14 (Khisamiev [48]). *Any strongly X -constructive abelian group has a strongly X -constructive divisible closure.*

In conclusion, we formulate an open question. Suppose every recursively enumerable definable group in a given quasivariety $\mathcal{M}$ of groups is constructivizable. Is $\mathcal{M}$ a quasivariety of abelian groups?

3 Direct sums of cyclic and quasicyclic p -groups

In this section we consider the problem of the existence of constructivizations for the class of groups indicated in the title. We recall some notions from abelian group theory. If the orders of all elements of a periodic group A are bounded, then their least common multiple is called the *period* of A . We say that a group element $g \neq 0$ has a finite p -height h if the equation $p^n x = g$ has a solution exactly when $n \leq h$. If the equation $p^n x = g$ has a solution for each n , then the p -height of g is ∞ . A group G is called a p -group if for each $g \in G$ there is an $n \in \omega$ such that the order of g is p^n . If a p -group has a finite period, then it cannot have any elements of infinite height. The following theorem is well known.

Theorem 3.1 (Prüfer). *Every countable abelian p -group without elements of infinite height is a direct sum of cyclic subgroups.*

Thus from Proposition 1.3, we obtain:

Corollary 3.2 *Every countable abelian p -group of finite period is strongly constructivizable.*

Let $A = \bigoplus \{\mathbb{Z}_{p^{n_i}} \mid i \in \omega\}$. The set

$$\chi(A) = \left\{ (m, k) \mid \exists i_1, \dots, i_k \left(\bigwedge_{1 \leq r \neq s \leq k} (i_r \neq i_s) \wedge n_{i_1} = \dots = n_{i_k} = m \right) \right\}$$

is called the *characteristic* of A .

Theorem 3.3 *Let $G = A \oplus \mathbb{Z}_{p^\infty}^\alpha$, where $\alpha \in \omega + 1$ and A is a direct sum of cyclic p -groups. G is strongly X -constructivizable if and only if the characteristic $\chi(A)$ is X -recursive*

Proof. Assume G is strongly constructivizable. Notice the following equivalence:

$$(m, k) \in \chi(A) \text{ if and only if there is a pure subgroup } \mathbb{Z}_{p^m}^k \text{ of } G.$$

The right half of this equivalence can be expressed by a Σ_2^0 -formula in the signature of abelian groups.

$$A_{pm}(x) = \left(px = 0 \wedge \exists y (p^{m-1}y = x) \right)$$

$$A_{pmk} = \exists x_1, \dots, x_k$$

$$\left(\bigwedge_{i=1}^k A_{pm}(x_i) \wedge \left(\bigwedge \left\{ \neg A_{p(m+1)} \left(\sum_{i=1}^k s_i x_i \right) \mid 0 \leq s_i < p, \sum s_i \neq 0 \right\} \right) \right)$$

Thus, by the decidability of $\text{Th}(G)$, the characteristic $\chi(A)$ is recursive. Sufficiency follows from Proposition 1.3. $\square$

An s -function is a function $f(i, x)$ for which the following conditions hold.

- (i) The function $\lambda x f(i, x)$ is non-decreasing for each i .
- (ii) For every i , $\lim_x f(i, x) = m_i$ exists.

If, in addition, $m_0 < m_1 < \dots$, then $f(i, x)$ is called an s_1 -function. Let

$$\bar{\rho}f = \left\{ (m, k) \mid \exists i_1, \dots, i_k \left(\bigwedge_{1 \leq r \neq s \leq k} (i_r \neq i_s) \wedge m_{i_1} = \dots = m_{i_k} = m \right) \right\}$$

Theorem 3.4 (Khisamiev [43]). *Let A be a direct sum of cyclic p -groups whose orders are unbounded. A is X -constructivizable if and only if the following conditions hold.*

- (i) *The characteristic $\chi(A) \in \Sigma_2^X$.*
- (ii) *There exists an X -recursive s_1 -function $f(i, x)$ such that $\bar{p}f \subseteq \chi(A)$.*

Proof. To prove necessity, let (A, ν) be the constructive group

$$\bigoplus \{\mathbb{Z}_{p^{n_i}} \mid i \in \omega\}.$$

From the proof of Theorem 3.3, it follows that $\chi(A) \in \Sigma_2^0$. We prove the existence of the function $f(i, x)$. Let m_0 be the least number such that $\nu m_0 \neq 0$ and $p\nu m_0 = 0$. We take

$$f(0, x) = \max\{f(0, x-1), s+1 \mid p^s \nu x = \nu m_0\}.$$

To define $f(1, x)$, we use the lexicographical order on the set of pairs of natural numbers. Find the least pair (i, k) such that $p^s \nu k = \nu i$, $p\nu i = 0$, $\nu i \notin (\nu m_0)$, and $s > f(0, x)$. We take $f(1, x) = s+1$. Similarly, we can find $f(2, x)$, etc.. Necessity is proved.

To prove sufficiency, let $M = \chi(A) \in \Sigma_2^0$. The set M is recursively enumerable with respect to $\emptyset^{(1)}$. Therefore there is a recursive function $\varphi(x, y)$ such that M is the domain of the function $g(x) = \lim_y \varphi(x, y)$. The desired group can be built in stages with the help of φ and f . If the function φ “makes a mistake”, i.e., $\varphi(x, y) > \varphi(x, y+1)$ for some pair (x, y) , then a cyclic summand, corresponding to this pair, is built with the help of f . $\square$

Theorem 3.5 *Let $A = B \oplus \mathbb{Z}_{p^\infty}^k$ be an abelian p -group with $k < \omega$ and B be a direct sum of cyclic p -groups. A is X -constructivizable if and only if B is X -constructivizable.*

Proof. Sufficiency is evident. Necessity can be proved as in Theorem 3.4. The only difference is that in the construction of f we should choose elements νm_0 and νi which do not belong to the subgroup $\mathbb{Z}_{p^\infty}^k$. $\square$

Theorem 3.6 (Khisamiev [44]). *Let $A = B \oplus \mathbb{Z}_{p^\infty}^\omega$ be an abelian p -group where B is a direct sum of cyclic groups. A is X -constructivizable if and only if the characteristic $\chi(B)$ is a Σ_2^X -set.*

Proof. Necessity follows from the proof of Theorem 3.3. We take $h(i, n) = n$. The proof of sufficiency is similar to the proof of Theorem 3.4 with f replaced by h . Here, the summands, built with the help of the function h , are isomorphic to the quasicyclic p -group. $\square$

Let $\lambda y f(x, y)$ be a non-decreasing function. If $\lim_y f(x, y)$ does not exist, then we say $\lim_y f(x, y) = \infty$.

Theorem 3.7 (Khisamiev [51]). *Let A be a direct sum of cyclic and quasicyclic p -groups. A is X -constructivizable if and only if there is an X -partial recursive ternary function $f_p(x, y)$, $p \in P$, satisfying the following conditions.*

- (i) *For every p, x, y, z with $y \leq x$, if $f_p(x, 0)$ is defined, then $f_p(y, z)$ is defined, and $f_p(y, z) \leq f_p(y, z + 1)$.*
- (ii) *$A \simeq \bigoplus \{\mathbb{Z}_{p^{m_{p,i}}} \mid p \in P, i \in \omega\}$, where $m_{p,i} = \lim_y f_p(i, y)$.*

The proof is similar to the proof of Theorem 3.4.

The definition of a divisible group was given in §2. A group which does not contain a non-zero divisible subgroup is called a *reduced group*. It is known that an arbitrary abelian group A is a direct sum of its reduced part R and its divisible part D , i.e., $A = R \oplus D$. This raises the question of whether R and D are (strongly) constructivizable if A is (strongly) constructivizable. The following examples show that the answer is negative.

Let S be a non-recursive recursively enumerable set of prime numbers.

Example 3.1 (Dobritsa, Goncharov [8]). Let $R_0 = \bigoplus \{\mathbb{Z}_p \mid p \in P \setminus S\}$ and $D_0 = \bigoplus \{\mathbb{Z}_{p^\infty} \mid p \in S\}$. It is easy to verify that R_0 is not constructivizable and the group $A_0 = R_0 \oplus D_0$ is constructivizable.

Example 3.2 Let $G_p = \bigoplus \{\mathbb{Z}_{p^n} \mid n \in \omega, n > 0\}$ and

$$A_1 = \bigoplus \{G_p \mid p \in P\} \oplus \bigoplus \{\mathbb{Z}_{p^\infty}^\omega \mid p \in P \setminus S\}.$$

It is easy to verify that A_1 is constructivizable, but that its divisible part D_1 is not constructivizable.

We consider this question restricted to abelian p -groups. A countable divisible p -group D can be decomposed into a direct sum of quasicyclic groups. By Proposition 1.3, the group D is strongly constructivizable. Therefore the divisible part of a p -group is constructivizable. We consider the reduced part of a p -group.

Proposition 3.8 (Khismiev [43]). *There is an $X^{(1)}$ -recursive set M such that for every X -recursive s_1 -function f , $\bar{p}f \notin M$.*

Therefore, from Theorem 3.6 we obtain:

Corollary 3.9 *There is an abelian p -group A such that*

- (i) *A can be decomposed into a direct sum of cyclic and quasicyclic groups,*
- (ii) *A is constructivizable, but its reduced part is not constructivizable.*

Corollary 3.10 *There is an abelian p -group A such that*

- (i) *A can be decomposed into a direct sum of cyclic and quasicyclic groups,*
- (ii) *A^ω is constructivizable, but the group A is not constructivizable.*

Proof. Let M be as in Proposition 3.8, and $A = \bigoplus \{\mathbb{Z}_{p^m} \mid m \in M\} \oplus \mathbb{Z}_{p^\infty}$. By Theorems 3.4 and 3.5, the group A is non-constructivizable, and by Theorem 3.6, the group A^ω is constructivizable. $\square$

Remark 3.1 From Theorem 3.3, it follows that Corollary 3.9 is not true for strongly constructivizable groups. By Theorem 3.5, it is not possible to replace ω by $k < \omega$ in Corollary 3.10.

4 Constructive abelian p -groups

In this section we consider the problem of the existence of (strong) constructivizations for abelian p -groups. According to Ulm's theorem (Fuchs [24, p. 63]), any countable abelian p -group A is fully defined by its Ulm sequence. The problem of the existence of (strong) constructivizations is solved below for abelian p -groups with finite Ulm length.

Recall the following definitions. Let A be a countable reduced abelian p -group. A^1 denotes the subgroup consisting of the elements of A with infinite height. If σ is an ordinal, then the σ -th Ulm subgroup A^σ of A is defined by: $A^0 = A$, $A^{\sigma+1} = (A^\sigma)^1$ and $A^\sigma = \bigcap_{\alpha < \sigma} A^\alpha$, when σ is a limit ordinal. The least ordinal number τ for which $A^\tau = 0$ is called the *Ulm length* $u(A)$ of A . For every $\sigma < \tau$, the factor group $A_\sigma = A^\sigma / A^{\sigma+1}$ is called the σ -th *Ulm factor* of A . The well-ordered sequence

$$A_0, A_1, \dots, A_\sigma, \dots \quad (\sigma < \tau)$$

is called the *Ulm sequence* of A . By Prüfer's theorem, every factor A_σ can be decomposed into a direct sum of cyclic p -groups. By Ulm's theorem, a countable reduced abelian p -group A is uniquely defined by its Ulm sequence.

The following theorem gives a sufficient condition for the constructivizability of a reduced p -group.

Theorem 4.1 (Dobritsa, Khisamiev, Nurtazin [9]). *Let A be a countable reduced abelian p -group for which the following hold.*

- (i) *The Ulm length $u(A)$ is equal to τ .*
- (ii) *There are constructive numerations π and ν_n of the ordinal τ and the Ulm factors $A_{\pi(n)}$ such that the sequence of constructive groups $\{(A_{\pi(n)}, \nu_n) \mid n \in \omega\}$ is computable.*

Then the group A has a constructivization ν such that the set

$$\{\langle n, m \rangle \mid \nu m \in A^{\pi(n)}\}$$

is recursive.

Theorem 4.2 (Dobritsa, Khisamiev, Nurtazin [9]). *If a reduced abelian p -group is constructivizable, then its Ulm length is a constructivizable ordinal.*

The following theorem gives a connection between the constructivizability of A and the constructivizability of both A^1 and A/A^1 .

Theorem 4.3 (Khisamiev [53, 54]). *Let A be a countable abelian p -group which is not a direct sum of cyclic and quasicyclic groups. A is (strongly) X -constructivizable if and only if the subgroup A^1 is $(X^{(1)}-)$ $X^{(2)}$ -constructivizable and the factor group $A/A^{(1)}$ is (strongly) X -constructivizable.*

Proof. To prove necessity, let (A, μ) be a constructive group. It is easy to verify that the subgroup A^1 is $\emptyset^{(2)}$ -recursive. We prove that the factor group $A_0 = A/A^1$ is constructivizable. By Theorem 3.4, it is sufficient to prove that the following holds.

- (i) The characteristic $\chi(A_0) \in \Sigma_2^0$.
- (ii) There is a recursive s_1 -function $f(i, x)$ such that $\bar{p}f \subseteq \chi(A_0)$.

Condition (i) follows from the proof of Theorem 3.3. To prove (ii), use the hypothesis that A cannot be decomposed into a direct sum of cyclic and quasicyclic groups. This implies that there is an element g of infinite height in the reduced part R of A such that, for every element h , the equality $ph = g$ implies that h has finite height. The construction of the required function $f(i, x)$ is similar to the proof of Theorem 3.4 with zero replaced by g .

To prove sufficiency, we need the following proposition.

Proposition 4.4 *Let (A^1, ν) be an $\emptyset^{(2)}$ -constructive abelian p -group, and $g(i, x)$ be a recursive s_1 -function. There is a constructivizable abelian p -group G such that*

$$(i) \quad G^1 \simeq A^1,$$

$$(ii) \quad G/G^1 \simeq \bigoplus \{\mathbb{Z}_{p^{m_i}} \mid i \in \omega\}, \text{ where } m_i = \lim_x g(i, x). \quad \square$$

The proof of this proposition is complicated, and therefore we omit it. To deduce the theorem from this proposition, suppose A satisfies the following conditions.

(i) The Ulm length of the reduced part R of A is more than 1.

(ii) The subgroup A^1 is $\emptyset^{(2)}$ -constructivizable.

(iii) The factor group $A_0 = A/A^1$ is constructivizable.

From (i), it follows that A_0 is a direct sum of cyclic p -groups of unbounded orders. By Theorem 3.4, we have:

$$(1) \quad \chi(A_0) \in \Sigma_2^1,$$

$$(2) \quad \text{there is a recursive } s_1\text{-function } f(i, x) \text{ such that } \bar{\rho}f \subseteq \chi(A_0).$$

We take

$$g(i, x) = f(2i, x), \quad h(i, x) = f(2i + 1, x),$$

$$\chi(A_0) = M, \quad M_0 = \bar{\rho}g, \quad M_1 = M \setminus M_0$$

It follows that $\bar{\rho}h \subseteq M_1$. As f is an s_1 -function, the following equivalence is true:

$$x \in M_1 \text{ iff } x \in M \quad \text{and} \quad \forall i \leq x + 1 \exists y \forall z \geq y (g(i, z) \neq x)$$

Hence $M_1 \in \Sigma_2^0$. By Theorem 3.4, there is a constructivizable group B_1 such that $\chi(B_1) = M_1$. By Proposition 4.4, there is a constructivizable group G such that conditions (i) and (ii) are true. The group $G \oplus B_1$ is constructivizable and isomorphic to A .

The proof of the theorem for the case of strong constructivizability is similar. $\square$

We give a number of corollaries to this theorem. In the corollaries, criteria for the (strong) constructivizability of p -groups with finite Ulm length are given. In what follows, instead of writing “strongly X -constructivizable” we write “ $X^{(-1)}$ -constructivizable”, and we let $X^{(0)}$ denote X .

Corollary 4.5 *Let A be a countable reduced abelian p -group with finite Ulm length $u(A) = n$. A is (strongly) X -constructivizable if and only if the Ulm factors A_i , $i < n$, are $(X^{(2i-1)}-)$ $X^{(2i)}$ -constructivizable groups.*

The proof is by induction on n .

Let $f_0(x, y)$ be an X -recursive function. $f_0(x, y)$ is called $X^{(-1)}$ -recursive if the following conditions hold.

- (i) $f_0(x, y)$ is independent of y .
- (ii) If $x_0 < x_1$ then $f_0(x_0, y) < f_0(x_1, y)$.

From Corollary 4.5 and Theorems 3.3 and 3.7, we obtain:

Corollary 4.6 *Let A be a countable reduced abelian p -group with finite Ulm length $u(A) = n$. A is (strongly) X -constructivizable if and only if there is a system of s -functions $\{f_i(x, y) \mid i < n\}$ such that*

- (i) *the function $f_i(x, y)$ is $(X^{(2i-1)}-)$ $X^{(2i)}$ -recursive,*
- (ii) *$A_i \simeq \bigoplus \{\mathbb{Z}_{p^{m_{ir}}} \mid r \in \omega\}$, where $m_{ir} = \lim_x f_i(r, x)$.*

Corollary 4.7 *Let $A = R \oplus \mathbb{Z}_{p^\infty}^\omega$ be an abelian p -group whose reduced part R has finite Ulm length $u(R) = n > 1$. A is (strongly) X -constructivizable if and only if the Ulm factors R_i , $i < n - 1$, of R are $(X^{(2i-1)}-)$ $X^{(2i)}$ -constructivizable, and the characteristic $\chi(R^{n-1})$ is a (Σ_{2n-1}^X-) Σ_{2n}^X -set.*

The proof is done by induction on n using Theorems 4.3, 3.3 and 3.6, and the following equalities for $r, s \in \omega$.

$$\Sigma_s^{X^{(r)}} = \Sigma_{s+r}^X, \quad X^{(s)^{(r)}} = X^{(s+r)}, \quad (R^1)_s = R_{s+1}.$$

Corollary 4.8 *Let $A = R \oplus \mathbb{Z}_{p^\infty}^k$, $k < \omega$, where R is the reduced part of A , and R has a finite Ulm length n . A is (strongly) X -constructivizable if and only if R satisfies the same conditions as in Corollary 4.7.*

The proof is done by induction on n using Theorems 4.3, 3.3 and 3.5.

In Section 5 we prove that in the case of strong constructivizability, Corollary 4.8 holds for any p -group.

Corollary 4.9 *Let A be a (strongly) X -constructivizable abelian p -group whose reduced part has infinite period. There are (strongly) X -constructivizable groups G and B_1 such that $A = G \oplus B_1$, and B_1 is a direct sum of cyclic p -groups of unbounded orders.*

The proof is similar to the deduction of Theorem 4.3 from Proposition 4.4.

Numerations ν and μ of a group A are called *autoequivalent* if there exists an automorphism φ of A and a recursive function f such that

$$\varphi \nu(n) = \mu f(n).$$

A group is called *autostable* if every two of its constructivizations are autoequivalent. It is not difficult to prove that the group B_1 of Corollary 4.8 is not autostable. Hence, from this corollary, we obtain that a constructivizable group which is not a direct sum of cyclic and quasicyclic p -groups is not autostable. The following theorem, which was proved independently by Goncharov and Smith, describes autostable p -groups.

Theorem 4.10 (Goncharov [28], Smith [87]). *An abelian p -group A is autostable if and only if either*

$$A \simeq \mathbb{Z}_{p^\infty}^\omega \oplus F \quad \text{or} \quad A \simeq \mathbb{Z}_{p^\infty}^\omega \oplus \mathbb{Z}_{p^n}^\omega \oplus F,$$

where F is a finite p -group, and $m, n \in \omega$.

Corollary 4.11 *Let A be an abelian group whose reduced part R has finite Ulm length. A finite direct power A^k of A , $k < \omega$, is (strongly) X -constructivizable if and only if A is (strongly) X -constructivizable.*

The proof is done by induction on the length $u(R) = n$ using Theorem 4.3. For $n = 1$, the corollary follows immediately from Theorems 3.3 and 3.6.

Dzgoev [13] showed that Corollary 4.11 is not true for torsion-free groups. By Corollary 3.10, it is impossible to replace k by ω in Corollary 4.11.

Corollary 4.12 *Let the Ulm length of the reduced part R of A be equal to $n < \omega$. If A is (strongly) X -constructivizable, then the factor group A/A^s is (strongly) X -constructivizable for any $s < n$.*

Indeed, let A be (strongly) X -constructivizable, $s < n$, and $A/A^s = B$. It is easy to verify that $B_i = A_i$ for $i \leq s - 1$. Hence from Corollaries 4.5, 4.7 and 4.8, B is (strongly) X -constructivizable.

Corollary 4.13 *Let A be a strongly X -constructivizable abelian p -group whose reduced part R has finite Ulm length. Then R is X -constructivizable.*

Indeed, let $A = R \oplus \mathbb{Z}_{p^\infty}^\omega$ and $u(R) = n$, $n < \omega$. By Corollary 4.7, the factors R_i , $i < n - 1$, are $X^{(2i-1)}$ -constructivizable, and the characteristic $\chi(r^{n-1}) \in \Sigma_{2n-1}^X$. Hence the group R^{n-1} is $X^{(2n-2)}$ -constructivizable. By Corollary 4.5, R is X -constructivizable.

From Proposition 3.8 and Corollary 4.7, it follows that there is a strongly constructivizable p -group such that the Ulm length $u(R)$ of the reduced part R is equal to 2, and R is not strongly constructivizable.

We note that the problem of necessary and sufficient conditions on Ulm invariants for a countable p -group to be (strongly) constructivizable is open.

5 Strong constructivity and Kulikov's basis

The concept of a p -basis introduced by Kulikov [57] has important meaning in the theory of abelian groups. In this section we present a connection between the constructivizability of groups with an algorithm for divisibility by p^n , $n \in \omega$, and the existence of a recursive p -basis. From this connection, a criterion for the strong constructivizability of an abelian p -group is obtained, and an application of this criterion is given.

Let A be an arbitrary abelian group and p any fixed prime number. A set $\{a_i \mid i \in I\}$ of elements of A not containing 0 is called p -independent if,

for any finite subset $\{a_1, \dots, a_k\}$ and for any positive integers $n_1, \dots, n_k$, $n_1 a_1 + \dots + n_k a_k \in p^r A$ implies $p^r | n_i$ ($i = 1, \dots, k$).

A maximal p -independent set of elements of A is called a *Kulikov basis* (or p -basis). The subgroup generated by some p -basis is called a *Kulikov subgroup*.

Let (A, ν) be a constructive group. If there is an algorithm which determines whether νm is divisible by p^n for any $n, m \in \omega$, then the pair (A, ν) is called a *constructive group with an algorithm for p^ω -divisibility*.

Proposition 5.1 (Khisamiev [45]). *If the pair (A, ν) is a constructive abelian group with a recursively enumerable Kulikov basis, then (A, ν) has an algorithm for p^ω -divisibility.*

Proof. Let the elements

$$b_0, b_1, \dots$$

be a recursively enumerable Kulikov basis. We show that the predicate $p^n |$ is recursive. The definition of a Kulikov basis implies that for an arbitrary element $x \in A$, there are $y \in A$ and integers $\alpha_0, \dots, \alpha_{m-1}$ such that

$$x + \alpha_0 b_0 + \dots + \alpha_{m-1} b_{m-1} = p^n y \quad (*)$$

The constructivity of (A, ν) and the recursive enumerability of the Kulikov basis implies that the element y and integers $\alpha_0, \dots, \alpha_{m-1}$ can be found effectively from x . From $(*)$ and the p -independence of $b_0, \dots, b_{m-1}$, we have that

$$p^n | x \iff \bigwedge_{i \leq m-1} (p^n | \alpha_i).$$

Hence the predicate $p^n |$ is recursive. $\square$

The following theorem shows that for abelian p -groups the converse also holds.

Theorem 5.2 (Khisamiev [44]). *Let (A, ν) be a constructive abelian p -group with an algorithm for divisibility. A has a recursive Kulikov subgroup B .*

This theorem is used to obtain the following criterion for the strong constructivizability of an abelian p -group in terms of generators and defining relations.

Theorem 5.3 (Khisamiev [44]). *An abelian p -group is strongly constructivizable if and only if it can be presented by generators $a_i, c_{j_n}, i < \alpha, j < \beta, n \in \omega$ and defining relations*

$$p^{n_i} a_i = 0, \quad p c_{j_0} = 0, \quad p c_{j_{n+1}} = c_{j_n} - b_{j_n}, \quad (1)$$

$$b_{j_n} = m_0 a_0 + \dots + m_{r_{j_n}} a_{r_{j_n}}, \quad (2)$$

where $\alpha, \beta \in \omega + 1$, the order of b_{j_n} is less than p^{n+2} , the functions $n(i) = n_i$, and $f(j, n) =$ the Gödel number of the sequence $(m_0, \dots, m_{r_{j_n}})$ are recursive, and the set

$$M = \left\{ \langle m, s \rangle \mid \exists i_1, \dots, i_s \left(\left(\bigwedge_{1 \leq k, r \leq s} k \neq r \rightarrow i_k \neq i_r \right) \wedge \left(\bigwedge_{1 \leq k \leq s} n_{i_k} = m \right) \right) \right\}$$

is recursive.

Proof. If a Kulikov subgroup B of A has finite period, then it is a direct summand of A , and all is proved. Therefore suppose that B does not have finite period. According to Theorem 5.2, there is a recursive Kulikov subgroup B of (A, ν) . Suppose that the rank $r(A/B)$ is infinite, as the case of $r(A/B) < \infty$ can be considered analogously. In B , we can effectively select elements $a_i, i \in \omega$, such that B is a direct sum of cyclic groups (a_i) , $B = \bigoplus \{(a_i) \mid i \in \omega\}$. The function $n'(i) = |a_i| = p^{n_i}$ is recursive. The decidability of the theory $\text{Th}(A)$ implies that the set M is recursive. To describe a procedure for the choice of $c_{j_0}, j \in \omega$, suppose the elements $c_{00}, \dots, c_{j_0}$ have been selected. There is an element c of order p with a least number such that the elements $c_{00}, \dots, c_{j_0}, c$ are linearly independent in the module B . We take $c_{(j+1)0} = c$. To describe the choice of $c_{j(n+1)}$ and b_{j_n} , suppose the elements c_{j_n} and $b_{j(n-1)}$ have been selected such that $p^{n+1} c_{j_n} = p^n b_{j(n-1)} = 0$. As A/B is a divisible group, there are elements $b \in B, d \in A$ such that $c_{j_n} + b = pd$. Since B is pure, the elements d and b can be selected such that $p^{n+2} d = p^{n+1} b = 0$. We take $c_{j(n+1)} = d$ and $b_{j_n} = b$. Since B is recursive, it follows that the function $f(j, n)$ is recursive. An arbitrary element $a \in A$ can be uniquely written in the form

$$a = s_1 a_{i_1} + \dots + s_m a_{i_m} + t_1 c_{j_1 n_1} + \dots + t_r c_{j_r n_r}, \quad (3)$$

where s_i, t_j are integers, $0 < t_j < p$, and the indices $i_1, \dots, i_m$ as well as $j_1, \dots, j_r$ are distinct. If the group $\overline{A}$ is defined by the generators and

defining relations (1) and (2), then the map $\varphi : a_i \rightarrow a_i$, $\varphi : c_{jn} \rightarrow c_{jn}$ can be extended to an isomorphism of $\overline{A}$ onto A . Necessity is proved.

Sufficiency follows from Proposition 5.1 and the uniqueness of the normal form (3) for each $a \in A$. $\square$

The set of elements $a_i, c_{jn}, i, j, n \in \omega$ of A defined in this theorem is called a *quasibasis* of A . The proof of the theorem implies that, given a quasibasis, every element $a \in A$ has a unique normal form (3). The induced numeration μ is called the numeration defined by the given quasibasis.

Corollary 5.4 *An arbitrary strongly constructive numeration of an abelian p -group is recursively equivalent to the numeration defined by some quasibasis.*

With the help of Theorem 5.3, we can make Corollary 4.8 stronger.

Theorem 5.5 (Khisamiev [44]). *Let D be the divisible part of an abelian p -group A , and suppose D has finite rank. A is strongly constructivizable if and only if its reduced part R is strongly constructivizable.*

Proof. Sufficiency is evident. We give a sketch of the proof of necessity. Let $A = R \oplus D$ be a strongly constructivizable group. A has the presentation defined in Theorem 5.2. Using the finiteness of the subgroup

$$D[p] = \{d \in D \mid pd = 0\},$$

we can define a presentation of the subgroup R similarly. To do this, replace some of the elements c_{jn} , $j = m_1, \dots, m_k$ by the suitable elements d_{sn} , $s = 1, \dots, q$. All that remains is to apply Theorem 5.3. $\square$

Remark 5.1 Corollary 3.9 implies that

- (1) Theorem 5.2 is not true for constructive abelian p -groups,
- (2) the condition of finiteness of rank is essential.

An abelian group A is called an R_p -group if for any $a \in A$ and for any number $n \in \omega$ which is not divisible by p , there is a unique solution of the equation $nx = a$.

Remark 5.2 Theorems 5.2, 5.3, 5.5 and Corollary 5.4 are true for torsion-free R_p -groups. Of course, in the case of torsion-free R_p -groups, the elements a_i , c_{jn} in Theorem 5.3 have infinite order. The proof of this result is similar to the proof for abelian p -groups.

An insignificant modification of the proof of Theorem 5.2 implies the following result for any abelian group.

Theorem 5.6 *An abelian group is constructivizable with an algorithm for p^ω -divisibility if and only if there is a constructive numeration μ of the group A such that*

- (i) *there is a recursively enumerable Kulikov basis $\{b_i \mid i \in I\}$ for (A, ν) ,*
- (ii) *there is a recursively enumerable set of elements $\{c_j \mid j \in J\}$ such that the set of cosets $\{c_j + B\}$ is maximally linearly independent in the factor group A/B , where $B = \text{gr}(\{b_i \mid i \in I\})$.*

6 Torsion-free abelian groups

In this section we examine the problems of existence and uniqueness of constructivizations for the class of torsion-free abelian groups. Mal'tsev [66] described constructivizable torsion-free abelian groups of rank 1. Kurosh [58] and Mal'tsev [65] established a correspondence between the class of all torsion-free abelian groups of finite rank and a class of matrices over the p -adic numbers for all primes p . Dobritsa [6] showed that a torsion-free abelian group A with finite rank is constructivizable if and only if the corresponding matrix can be given effectively.

6.1 Countable constructivizable groups

Here we give a description of countable constructivizable torsion-free abelian groups in terms of partial recursive functions (Khisamiev [52]). Recall that P is the set of all prime numbers, and p_i is the i -th prime number. Given a partial recursive function $f(s, i, n)$ we define a presentation of an abelian group A_f in steps.

STEP t : Let $t = \langle s, i, n \rangle$, $p = p_s$. Do t steps in the calculation of $f(s, i, n)$. The following cases are possible.

(i) $f(s, i, n)$ is defined. Consider the subcases.

- (1) $n = 0$ and the generators a_{j0} , $j < i$, were introduced before step t . Introduce the generators a_{i0} , $\{a_{i0}^{(q)} \mid q \in P\}$ and the defining relations $\{a_{i0}^{(q)} = a_{i0} \mid q \in P\}$.
- (2) $i = 0$, $n > 0$ and a generator $a_{0(n-1)}^{(p)}$ was introduced before step t . Introduce the generator $a_{0n}^{(p)}$ and defining relation $a_{0(n-1)}^{(p)} = pa_{0n}^{(p)}$.
- (3) $i > 0$, $n > 0$ and $f(s, i, n) = \langle \xi_0, \dots, \xi_{i-1} \rangle$, $\xi_j = \langle \alpha_j, m_j \rangle$, $0 \leq \alpha_j < p$, $j \leq i-1$ and the generators $a_{i(n-1)}^{(p)}$, $a_{jm_j}^{(p)}$ were introduced before step t . Introduce the generator $a_{in}^{(p)}$ and the defining relation

$$a_{i(n-1)}^{(p)} + \alpha_0 a_{0m_0}^{(p)} + \dots + \alpha_{i-1} a_{(i-1)m_{i-1}}^{(p)} = pa_{in}^{(p)} \quad (*)$$

(ii) If none of the cases (i) (1)–(i) (3) are satisfied, then we introduce nothing.

Step t is finished, and we pass to the next step.

The construction of the group A_f is finished. From this construction, we can define the constructive numeration ν_f of A_f in a natural way.

Theorem 6.1 (Khisamiev [52]). *A countable torsion-free abelian group A is X -constructivizable if and only if there exists an X -partial recursive function $f(s, i, n)$ such that A and A_f are isomorphic.*

Proof. Sufficiency is easy because of the construction of the group A_f . We prove necessity. Let A be a constructivizable group with countable rank. The case when A has finite rank is similar. By Theorem 2.2, there exists a constructivization ν of A such that (A, ν) has a recursively enumerable basis

$$b_0, b_1, \dots$$

We define the function $f(s, i, n)$. Let $p_s = p$. For every $s, i \in \omega$ we take $f(s, i, n) = 0$ and $b_{i0}^{(p)} = b_i$. If $i = 0$ then we take

$$f(s, 0, n+1) = \begin{cases} 1 & \text{if } A \models \exists x(p^{n+1}x = b) \\ \text{undefined} & \text{otherwise} \end{cases}$$

If $f(s, 0, n+1)$ is defined, then for any $k \leq n+1$, let $b_{0k}^{(p)}$ denote the element from A such that $p^k b_{0k}^{(p)} = b_0$.

Let $n, i > 0$. We define the value of $f(s, i, n)$ and possibly the element $b_{in}^{(p)}$ in steps. Suppose that t steps are done, and the values $f(p, j, y)$, $y \leq t_j$, $j \leq i$, $t_i = n-1$, $n \leq t$, are found and the elements $b_{jr}^{(p)}$, $r \leq t_j$, are defined such that the relations of form $(*)$ have been satisfied.

STEP $t+1$: Let $t+1 = \langle s, i \rangle$. We check for the existence of numbers x, α_j, m_j , $j \leq i-1$, such that $x \leq t$, $\alpha_j < p$, $m_j \leq t_j$ and the following equation is satisfied in the group (A, ν) .

$$b_{in}^{(p)} + \alpha_0 b_{0m_0}^{(p)} + \cdots + \alpha_{i-1} b_{(i-1)m_{i-1}}^{(p)} = p \nu x \quad (**)$$

- (i) Suppose that such numbers exist. Choose α_j, m_j from among them such that the sequence $\theta = (\alpha_{i-1}, \dots, \alpha_0, m_{i-1}, \dots, m_0)$ is the least with respect to the lexicographical ordering on the set of all finite sequences of natural numbers. For this θ , let $b_{in}^{(p)}$ denote the element νx such that $(**)$ holds, and we take

$$f(s, i, n) = \langle \xi_0, \dots, \xi_{i-1} \rangle, \quad \text{where } \xi_j = \langle \alpha_j, m_j \rangle.$$

- (ii) Suppose that there are no such numbers. The value of $f(s, i, n)$ and the element $b_{in}^{(p)}$ are not defined.

Step $t+1$ is finished. We pass to the next step.

The function f is defined. From this function, we define the group A_f as above. We can prove that the groups A_f and A are isomorphic. $\square$

Theorem 6.2 *Let (A, ν) be a constructive torsion-free abelian group with an algorithm for linear independence. We can uniformly find the index e of a partial recursive function f such that the constructive groups (A, ν) and (A_f, ν_f) are recursively isomorphic.*

The proof is similar to the proof of Theorem 6.1.

Remark 6.1 Let $\psi_e(x, y, z)$ be the partial recursive function with index e . From this function, we defined the constructivizable torsion-free group with an algorithm for linear independence $(A_e, \nu_e) = (A_{\psi_e}, \nu_{\psi_e})$. This defines a

computable numeration α of a class of torsion-free constructivizable groups. From Theorems 6.1 and 6.2, it follows that α is a numeration for the class of all constructivizable torsion-free groups with an algorithm for linear independence.

6.2 Autostable groups

The following theorem gives a description of the autostable torsion-free groups.

Theorem 6.3 (Nurtazin [76]). *A torsion-free abelian group is autostable if and only if it has finite rank.*

Dobritsa [7] built a non-autostable group A such that

- (i) the periodic part T of A is autostable,
- (ii) the rank of A/T is equal to 1.

The problem of finding algebraic criteria for the autostability of abelian groups is open.

7 Model theoretic method in the theory of constructivizable abelian groups

In §2–§6, we considered the problem of determining for various classes of abelian groups, which groups from each class are (strongly) constructivizable. In this section, we consider a slightly different problem.

Let L be a class of elementary theories of abelian groups. Which theories from L have (strongly) constructive models? This problem has been examined for several classes of theories. It is known (Ershov [16]) that any decidable theory is defined by its strongly constructive models. From here, for example, it follows that any theory of Boolean algebras has a strongly constructive model. Ershov [17] proved that any recursively enumerable theory with finite barriers has a constructive model. In particular, any recursively enumerable extension of the theory of trees has a constructive model. A similar result for linearly ordered sets was proved earlier by Peretyat'kin [79]. Lerman and Schmerl [62] made this result stronger by replacing the recursively enumerable extension by a Σ_2^0 -extension. It is impossible, however, to

strengthen the result obtained for the theory of trees. In §7.1, we examine this problem for the class of abelian groups. We obtain a criterion for the existence of a constructive model for a complete theory of abelian groups. We prove that any Σ_2^0 -theory of torsion-free abelian groups has a constructive model. We also give a recursively enumerable theory of abelian groups that does not have a constructive model.

In the study of elementary theories, properties of special models, for example, prime or saturated models, play an important role. In the context of the theory of constructive models, these questions have been examined in a series of papers. Goncharov [29] and Goncharov and Nurtazin [34] gave criteria for the (strong) constructivizability of a prime model. Goncharov [27] and Peretyat'kin [80] obtained criteria for the strong constructivizability of a homogeneous model. Morozov [74] proved that any countable saturated Boolean algebra is strongly constructivizable. From a theorem about the strong constructivizability of a prime model of a theory with a strongly constructivizable saturated model (Goncharov [27]), it follows that any countable prime Boolean algebra is strongly constructivizable (Mead [69]). Morozov [73] described strongly constructivizable homogeneous Boolean algebras. The questions of the (strong) constructivizability of prime and saturated models of theories of abelian groups are discussed in §7.2.

7.1 Complete theories

Let G be a group, $G[p^n] = \{x \in G \mid p^n x = 0\}$. The Szmielew invariants of G will be defined with the help of the following formulae.

- (1) $G \models A_{pnk}$ iff G contains $\mathbb{Z}_{p^n}^k$ as a pure subgroup. (The formulae A_{pnk} were used in the proof of Theorem 3.3.)
- (2) $G \models B_{pnk}$ iff $\mathbb{Z}_{p^n} \leq G$.
- (3) $G \models C'_{pnk}(g_1, \dots, g_k)$ iff the elements $g_1, \dots, g_k \in G$ have the following property: the images $g'_1, \dots, g'_k$ of $g_1, \dots, g_k$ in the factor group

$$\overline{G} = G/G[p^n]$$

are such that $g'_1 + p\overline{G}, \dots, g'_k + p\overline{G}$ are linearly independent in $\overline{G}/p\overline{G}$.

- (4) $C_{pnk} = \exists x_1, \dots, x_k C'_{pnk}(x_1, \dots, x_k)$.

The following numbers are called the *Szmielew invariants* of G .

$$\alpha_{pn}(G) = \sup \{k \in \omega \mid G \models A_{pnk}\},$$

$$\beta_p(G) = \inf \{ \sup \{k \in \omega \mid G \models B_{pnk}\} \mid n \in \omega \},$$

$$\gamma_p(G) = \inf \{ \sup \{k \in \omega \mid G \models C_{pnk}\} \mid n \in \omega \}.$$

Szmielew [89] proved that two groups G and H are elementarily equivalent if and only if these invariants of G and H are the same.

Let T be a complete theory of groups. We let

$$A(T) = \{ (p^{n+1}, k) \mid A_{pnk} \in T \},$$

$$B(T) = \{ (p^{n+1}, k) \mid B_{pnk} \in T \},$$

$$C(T) = \{ (p, k) \mid \exists N_p [\forall n \geq N_p (\neg A_{pn1} \in T) \wedge C_{pN_p k} \in T] \},$$

$$P(T) = \{ p \mid \exists N_p [\forall n \geq N_p (\neg A_{pn1} \in T)] \}.$$

A set C' of pairs (p, k) , $k > 0$, is called a T -extension of the set $C(T)$ if

- (i) $C(T) \subseteq C'$,
- (ii) if $(p, k) \in C' \setminus C(T)$ then $p \notin P(T)$,
- (iii) if $(p, k) \in C'$ and $0 < s < k$ then $(p, s) \in C'$.

Theorem 7.1 (Khisamiev [47]). *A complete theory T of abelian groups has an X -constructivizable model if and only if the following hold.*

- (i) $A(T) \in \Sigma_2^X$,
- (ii) $B(T) \in \Sigma_1^X$, and
- (iii) The set $C(T)$ has a T -extension $C' \in \Sigma_2^X$.

We will sketch the proof below, but first we state two propositions of independent interest. Let A_p, B_p be sets of pairs (p^n, s) , $n, s > 0$, $p \in P$. Define

$$A_{p^n} = \{s \mid (p^n, s) \in A_p\} \quad \text{and} \quad \alpha_{pn} = \max A_{p^n}$$

$$B_{p^n} = \{s \mid (p^n, s) \in B_p\} \quad \text{and} \quad \beta_{pn} = \max B_{p^n}$$

Suppose that the following conditions hold.

- (i) If $(p^n, s) \in A_p$ and $1 < r < s$, then $(p^n, r) \in A_p$. Similarly for B_p .
- (ii) $\beta_{pn} = \alpha_{pn} + \beta_{p(n+1)}$.

We take $\mathbb{Z}_{p^n}^0 = 0$, $\mathbb{Z}_{p^n}^\infty = \mathbb{Z}_{p^n}^\omega$, and $\mathbb{Z}_{p^0} = 0$.

Proposition 7.2 *Let $A_p \in \Sigma_2^X$ and $B_p \in \Sigma_1^X$ be such that conditions (i) and (ii) hold. We can build an X -constructivizable p -group*

$$G_p = \bigoplus \{ \mathbb{Z}_{p^n}^{\alpha_{pn}} \mid n \in \omega \} \oplus \mathbb{Z}_{p^\infty}^{\xi_p}, \quad \xi_p \in \omega \cup \{ \infty \}$$

such that

$$\beta_{pn} = \sum_{i=0}^{\infty} \alpha_{p(n+1)} + \xi_p.$$

This construction can be done uniformly in the Σ_2^X and Σ_1^X indices of A_p and B_p .

In order to formulate the next proposition, we need some notation. Let R_p denote the additive group of rational numbers whose denominators are prime relative to p . Let C' be a set of pairs (p, s) , $s > 0$, $p \in P$, such that if $(p, s) \in C'$, $0 < r < s$, then $(p, r) \in C'$. We take

$$C'_p = \{ s \mid (p, s) \in C' \}, \quad \gamma'_p = \max C'_p,$$

$$H(C') = \bigoplus \{ R_p^{\gamma'_p} \mid p \in P \} \oplus \mathbb{Q}^\omega.$$

Proposition 7.3 *The group $H(C')$ is X -constructivizable if and only if $C' \in \Sigma_2^X$.*

If the set of sentences of a theory T is Σ_2^X , then T is called a Σ_2^X -theory.

Corollary 7.4 *Any Σ_2^X -theory of torsion-free abelian groups has an X -constructive model.*

Indeed, one can define the set

$$C = \{ (p, s) \mid T \models C_{p1s} \}.$$

Because T is a Σ_2^0 -theory, it follows that $C \in \Sigma_2^0$. According to Proposition 7.3, the group $H(C)$ is constructivizable. From the criterion of elementary equivalence of groups, it is easy to obtain that $H(C)$ is a model for the theory T . $\square$

Now we prove the sufficiency of the conditions in the theorem. Let the sets $A(T) = A$, $B(T) = B$ and C' be as in the theorem. Let

$$A_p = \{(p^n, s) \mid (p^n, s) \in A\} \quad \text{and} \quad B_p = \{(p^n, s) \mid (p^n, s) \in B\}.$$

Let ξ_p be a number such that

$$\beta_{p(n+1)} = \sum_{i=0}^{\infty} \alpha_{p(n+1+i)} + \xi_p.$$

It is easy to verify that the conditions of Propositions 7.2 and 7.3 hold for the sets A_p , B_p and C' . Therefore the group

$$G = \bigoplus \{G_p \mid p \in P\} \oplus \mathbb{Q}^\omega$$

where

$$G_p = \bigoplus \{\mathbb{Z}_{p^{n+1}}^{\alpha_{pn}} \mid n \in \omega\} \oplus \mathbb{Z}_p^{\xi_p} \oplus R_p^{\gamma'_p}$$

is constructivizable. From a computation of the Szmielew invariants of G , it follows that G is a model of T .

Necessity follows from the definitions of the sets $A(T)$, $B(T)$ and C' , and the formulae A_{pnk} , B_{pnk} and C_{pnk} . $\square$

We indicate a number of corollaries of this theorem.

Corollary 7.5 *If an abelian group G is elementarily equivalent to its periodic part $T(G)$, then the theory $T = \text{Th}(G)$ has an X -constructive model if and only if $A(T) \in \Sigma_2^X$ and $B(T) \in \Sigma_1^X$.*

Indeed, in this case $C(T) = \emptyset$, and therefore condition (iii) of Theorem 7.1 is satisfied.

Corollary 7.6 *Let T be a complete theory of abelian groups such that $P(T) \in \Sigma_2^X$. T has an X -constructive model if and only if $B(T) \in \Sigma_1^X$ and $A(T), C(T) \in \Sigma_2^X$.*

Proof. Sufficiency follows immediately from Theorem 7.1. We prove necessity. Suppose T has a constructive model. According to Theorem 7.1, the set $C(T)$ has a T -extension $C' \in \Sigma_2^0$. Therefore

$$(p, k) \in C(T) \quad \text{iff} \quad p \in P(T) \quad \text{and} \quad (p, k) \in C'.$$

The right half of this equivalence defines a Σ_2^0 -set. Hence $C(T) \in \Sigma_2^0$. $\square$

Corollary 7.7 *A complete theory T of torsion-free abelian groups has an X -constructive model if and only if*

$$C_0(T) = \{(p, k) \mid C_{p1k} \in T\} \in \Sigma_2^X.$$

Proof. Because T is a theory of torsion-free groups, $A(T) = B(T) = \emptyset$ and $P(T)$ coincides with the set of all prime numbers. By Corollary 7.6, T has a constructive model if and only if $C(T) \in \Sigma_2^0$. It is easy to verify that for a theory of torsion-free groups, $C_0(T) = C(T)$. $\square$

Khisamiev [47] built a recursively enumerable theory of abelian groups having no constructive model. It was shown there that the conditions of Theorem 7.1 are independent.

7.2 Prime and saturated models

A model M is called prime if it can be elementarily embedded in any elementarily equivalent model. Let $\{p_0, \dots, p_{n-1}\}$ be a finite set of prime numbers. Let $R_{p_0 \dots p_{n-1}}$ denote the additive group of rational numbers whose denominators are prime relative to $p_0, \dots, p_{n-1}$. The following two theorems give a description of the prime model for theories of abelian groups.

Theorem 7.8 (Deissler [4]). *A countable torsion-free abelian group G is prime if and only if G is isomorphic to $R_{p_0 \dots p_{n-1}}$ for some set $\{p_0, \dots, p_{n-1}\}$ of prime numbers.*

Proposition 7.9 *An abelian p -group G is prime if and only if there exist numbers $\alpha_n, \beta \in \omega + 1$ such that*

- (i) *if the set $\{n \mid \alpha_n \neq 0\}$ is infinite, then $\beta = 0$,*
- (ii) $G = \bigoplus \{\mathbb{Z}_{p^{n+1}}^{\alpha_n} \mid n \in \omega\} \oplus \mathbb{Z}_{p^\infty}^\beta.$

Theorem 7.10 (Molokov [72]). *A finite or countable abelian group G is prime if and only if one of the following conditions is satisfied.*

- (i) G is a periodic group whose p -components are prime,
- (ii) G is a direct sum of a group with a finite period, a periodic divisible group, and a prime torsion-free group which is not isomorphic to $\mathbb{Q}$,
- (iii) G is a direct sum of a group with finite period and a group which is not isomorphic to $\mathbb{Q}$.

From Propositions 1.1 and 1.3, and Theorem 7.8, we obtain

Corollary 7.11 *If a complete theory T of torsion-free abelian groups has a prime model G , then G is strongly constructivizable.*

Corollary 7.12 *Let T be a complete theory of abelian groups with a constructivizable model H and a prime model G . If for any prime number p the set $\{n \mid \alpha_{p^n}(G) \neq 0\}$ is finite, then G is constructivizable.*

Proof. If either condition (ii) or (iii) from Theorem 7.10 holds for G , then G is constructivizable by Corollary 7.11. Therefore we suppose that condition (i) from Theorem 7.8 is satisfied. From Proposition 7.9, we have

$$G = \bigoplus \{G_p \mid p \in P\}$$

where

$$G_p = \bigoplus \{\mathbb{Z}_{p^{n+1}}^{\alpha_{pn}(G)} \mid n \in \omega\} \oplus \mathbb{Z}_{p^\infty}^{\beta_p(G)}.$$

Let H be a countable model of the theory T . From the elementary equivalence of G and H , and from condition (i), it follows that the periodic part $T(H)$ of H is isomorphic to G . By Corollary 1.5, the subgroup $T(H)$ is constructivizable. Hence G is constructivizable. $\square$

It turns out that there exists a countable prime group which is not constructivizable.

Proposition 7.13 (Khisamiev [47]). *There exists a complete theory T of abelian groups such that T has a constructive model H and a prime model G , but G is not constructivizable.*

Proof. According to Corollary 3.9, there exists a constructivizable p -group $H = G \oplus \mathbb{Z}_{p^\infty}^\omega$ such that the subgroup G can be decomposed into a direct sum of cyclic p -groups, and G is not constructivizable. Let $T = \text{Th}(H)$. It is easy to verify that G is a prime model of T . $\square$

We consider the questions of the strong constructivizability of prime and saturated models of theories of abelian groups.

Proposition 7.14 *Let*

$$G = \bigoplus \{G_p \mid p \in P\} \quad \text{where} \quad G_p = \bigoplus \{\mathbb{Z}_{p^{n+1}}^{\alpha_{pn}} \mid n \in \omega\} \oplus \mathbb{Z}_{p^\infty}^{\beta_p}.$$

and for any prime number p the following condition is satisfied.

- (i) *If the set $\{n \mid \alpha_{pn} \neq 0\}$ is infinite, then $\beta_p = 0$.*

G is strongly constructivizable if and only if $\text{Th}(G)$ is decidable.

Proof. It is evident that from the strong constructivizability of G , it follows that $\text{Th}(G)$ is decidable. Suppose $\text{Th}(G)$ is decidable. Then the sets $A(T)$ and $B(T)$ defined at the beginning of §7.1 are recursive. Therefore, from (i), we have the following property: for any $p \in P$ there is a strong constructivization ν_p of G_p such that the sequence of pairs $\{(G_p, \nu_p) \mid p \in P\}$ is computable. By Proposition 1.3, the group is strongly constructivizable. $\square$

Corollary 7.15 (Khisamiev [47]). *Let T be a complete theory of abelian groups with a prime model G . The following conditions are equivalent.*

- (i) *G is strongly constructivizable.*
 (ii) *The theory T is decidable.*

Proof. It is evident that (ii) follows from (i). Suppose T is decidable. If condition (i) from Theorem 7.10 holds for G , then the corollary follows from Proposition 7.14. Suppose G satisfies either condition (ii) or (iii) from Theorem 7.10. There exists a finite sequence $p_0, \dots, p_{s-1}$, $s \geq 0$, of prime numbers, and $\beta_p, \gamma \in \omega + 1$, $p \in P$, such that

$$G = H \oplus \bigoplus \{\mathbb{Z}_{p^\infty}^{\beta_p} \mid p \in P\} \oplus R_{p_0 \dots p_{s-1}}^\gamma$$

where the subgroup H has finite period. The subgroups H and $R_{p_0 \dots p_{s-1}}^\gamma$ are strongly constructivizable. From the definition of the set $B(T)$ at the beginning of §7.1, we have

$$n < \beta_p \quad \text{iff} \quad (p^{m+1}, n) \in B(T) \quad (*)$$

As the theory T is decidable, the set $B(T)$ is recursive. Therefore, according to Propositions 1.1 and 1.3, it follows from $(*)$ that the group

$$\bigoplus \{ \mathbb{Z}_{p^\infty}^{\beta_p} \mid p \in P \}$$

is strongly constructivizable. Hence G is strongly constructivizable. $\square$

Proposition 7.16 (Khisamiev [47]). *Let T be a complete theory of abelian groups with a saturated model. The following conditions are equivalent.*

- (i) *There exists a constructivizable group of the theory T .*
- (ii) *All countable models of T are strongly constructivizable.*
- (iii) *The theory T is decidable.*

Proof. Let G be a saturated model of T . If G has a finite period, then T is countably categorical, and our assertion is evident. The periodic part of G is denoted by $T(G)$. It is not difficult to show (Eklof and Fischer [14]) that the reduced part $RT(G)$ of the subgroup $T(G)$ has a finite period, and $G/T(G)$ is a torsion-free divisible group. Therefore, from the completeness of T , it follows that any countable model A of T is isomorphic to a group $A_n = T(G) \oplus \mathbb{Z}^n$ for some $n \in \omega + 1$. Now we prove that (ii) follows from (i). Let $n \in \omega + 1$ be such that the group A_n is constructivizable. The prime model A_0 of T is isomorphic to $T(G)$, and hence A_0 is constructivizable. As

$$A_0 \simeq RT(G) \oplus \bigoplus \{ \mathbb{Z}_{p^\infty}^{\beta_p} \mid p \in P \}$$

where $RT(G)$ has a finite period, the theory T has a recursively enumerable set of axioms. Therefore T is decidable. It is easy to check that A_0 has a constructivization with an algorithm for divisibility. Hence by Proposition 1.1, A_0 is strongly constructivizable. By Proposition 1.3, the groups A_n , $n \in \omega + 1$ are also strongly constructivizable. Hence (ii) follows from (i). It is evident that (iii) follows from (ii). By Corollary 7.15, (iii) implies (i). $\square$

The following problem is open. Find necessary and sufficient conditions, similar to the conditions of Theorem 7.1, for a theory of abelian groups with a recursively enumerable set of axioms to have a constructive model.

8 Connections between constructivizability and strong constructivizability

It is known (Ershov [19, p. 316]) that every constructive model of a model-complete decidable theory is strongly constructive. This raises the question of which decidable theories have the property that every constructive model is strongly constructive. Goncharov [25, 26] solved this problem for the class of Boolean algebras. Khisamiev [42] proved that every constructivizable ordinal is strongly constructivizable. Here we consider the problem for complete decidable theories of abelian groups.

In §7, the Szmielew invariants $\alpha_{pn}(A)$, $\beta_p(A)$ and $\gamma_p(A)$ of a groups A were defined.

Theorem 8.1 (Khisamiev [45]). *Let A be an abelian group with a decidable theory. If one of the following conditions is satisfied,*

- (i) $\exists p(\{n \mid \alpha_{pn}(A) \neq 0\} \text{ is infinite}),$
- (ii) $\{p \mid \gamma_p(A) \neq 0\} \text{ is infinite},$
- (iii) $\exists p(\gamma_p(A) = \omega),$

then there is a constructivizable but not strongly constructivizable abelian group $\bar{A}$ which is elementarily equivalent to A . Otherwise, the constructivity of A implies strong constructivity.

Proof. Suppose condition (i) is satisfied. Let q be a prime number such that the set $\{n \mid \alpha_{qn} \neq 0\}$ is infinite. We can assume that the group A is countable. Let A_p be the p -component of A , and B_p be a Kulikov subgroup of A_p . The decidability of the theory $T = \text{Th}(A)$ implies that the characteristic $\chi(B_q) = M_0$ of B_q is an infinite recursive set. By Proposition 3.8, there is a $\emptyset^{(2)}$ -recursive set M'_1 such that for every $\emptyset^{(1)}$ -recursive s_1 -function $f(i, x)$, the set $\bar{p}f$ is not contained in $M_1 = \{(m, 1) \mid m \in M'_1\}$. Let C be a q -group with Ulm length $u(C) = 2$ and Ulm factors C_0, C_1 with $\chi(C_0) = M_0$,

$\chi(C_1) = M_1$. It is evident that the subgroup C^1 of C is $\emptyset^{(2)}$ -constructivizable. By Corollary 4.5, C is constructivizable. By Theorem 3.4, C^1 is not $\emptyset^{(1)}$ -constructivizable. Hence from Theorem 4.3, it follows that C is not strongly constructivizable. Let

$$\overline{A} = C \oplus \bigoplus \{B_p \oplus \mathbb{Z}_{p^\infty}^{\beta_p(A)} \oplus R_p^{\gamma_p(A)} \mid p \in P \setminus \{q\}\}. \quad (*)$$

The definition of the group R_p is given in §7. It is easy to verify that the Szmielew invariants of A and $\overline{A}$ are the same. Therefore these groups are elementarily equivalent, $A \equiv \overline{A}$. From the decidability of $\text{Th}(A)$, it follows that the last summand in the right part of $(*)$ is a constructivizable group. Therefore the group $\overline{A}$ is constructivizable. The Szmielew invariants of C are effective computable from the Szmielew invariants of $\overline{A}$. Therefore, the decidability of $\text{Th}(C)$ follows from the decidability of $\text{Th}(\overline{A})$. Since the strong constructivizability of $\overline{A}$ implies the strong constructivizability of C , and since C is not strongly constructivizable, it follows that $\overline{A}$ is not strongly constructivizable. The theorem is proved for case (i).

For case (ii), we need two further results. Let G be a torsion-free abelian group with rank $r(G) = 1$. We can assume that $G \leq \mathbb{Q}$ and $1 \in G$. The set

$$\chi(G) = \{(p, n) \mid n \geq 1, 1/p^n \in G\}$$

is called the characteristic of G .

Proposition 8.2 (Mal'tsev [67]). *G is constructivizable if and only if its characteristic $\chi(G)$ is recursively enumerable.*

Proposition 8.3 *Let G be a torsion-free abelian group, S be an infinite recursive set of prime numbers, and $G = \bigoplus \{R_p \mid p \in S\}$. There is a constructivizable group $\overline{G}$ with $r(\overline{G}) = 1$ such that $\overline{G} \equiv G$ and $\overline{G}$ is not a pure subgroup of any strongly constructive group.*

Proof. Let S_0 be a recursively enumerable but non-recursive subset of S . Let

$$M = \{(p, 1) \mid p \in S\} \cup \{(p, 2) \mid p \in S_0\} \cup \{(p, n) \mid n \geq 1, p \notin S\}.$$

Let $\overline{G}$ be a torsion-free group with $r(\overline{G}) = 1$ such that $\chi(\overline{G}) = M$. It is easy to verify that $\overline{G}$ is the required group. $\square$

Now we consider case (ii). In this case, the set $S = \{p \mid \gamma_p(A) \neq 0\}$ is infinite. From the decidability of $\text{Th}(A)$, it follows that S is recursive. Let

$$C_0 = \bigoplus \{R_p \mid p \in S\}$$

and the group $\overline{C}_0$ be built from C as in Proposition 8.3. Let

$$\overline{A} = \bigoplus \{H_p \mid p \in P\} \oplus C_0$$

where

$$H_p = \bigoplus \{\mathbb{Z}_{p^{n+1}}^{\alpha_{pn}(A)} \mid n \in \omega\} \oplus \mathbb{Z}_{p^\infty}^{\beta_p(A)} \oplus R_p^{\gamma_p(A)}.$$

Using Proposition 8.3, it is easy to prove that $\overline{A}$ is the required group. This proves the theorem for case (ii).

Before considering case (iii), we recall some definitions. A subgroup B of a group A is called p -pure if for every $b \in B$ the p -height of b in B is equal to the p -height of b in A . Let $n \in \omega$, $n \geq 1$, be a fixed number. If the predicate $p^n \mid$ in a numerable group (A, ν) is recursive, then the pair (A, ν) is called a group with an algorithm for divisibility by p^n .

Proposition 8.4 *There is a constructivizable torsion-free abelian group which is not a p -pure subgroup of any constructivizable group which has an algorithm for divisibility by p , and whose p -subgroup is divisible.*

Proof. The values of all functions defined below are assumed to belong to the set $\{0, 1, \dots, p-1\}$. Let $K(i, n)$ be the universal function for the class of all 1-place partial recursive functions. Suppose that if the value $K(i, n)$ is not defined, then $K(i, n+1)$ is also not defined. We use $K(i, n)$ to build the group $A = A(K)$.

We build the generators and the defining relations for every i in steps. The symbol g is an element of the set of generators. Suppose $K(i, x)$ for $x \leq t$ have been calculated for t steps, and the values $K(i, 0), \dots, K(i, n(t)-1)$ are known. Here, $n(t)$ is a function which depends on t . Suppose the generators $c_{i0}, \dots, c_{in}$ and the defining relations

$$c_{im} + K(i, m)g = p c_{i(m+1)} \quad (m \leq n-1)$$

have been introduced.

STEP $t+1$: We calculate $K(i, n)$ for $t+1$ steps, and consider the following cases.

- (i) If the value of $K(i, n)$ is defined, then we introduce the generator $c_{i(n+1)}$ and the relation

$$c_{in} + K(i, n)g = p c_{i(n+1)}.$$

- (ii) If the value of the function is not defined, then we introduce nothing.

It is not difficult to prove that this is the required group. $\square$

Corollary 8.5 *There is a constructivizable torsion-free abelian group A and a prime number p such that A does not have a constructivization with an algorithm for divisibility by p .*

Now we consider case (iii). Let A be a group with a decidable theory, and $\gamma_q(A) = \omega$ for some prime number q . We assume that condition (i) is not satisfied. Hence the sets

$$M_p = \{n \mid \alpha_{pn}(A) \neq 0\}$$

are finite. In §5, we gave the definition of a torsion-free R_p -group. Let the group G be built according to Proposition 8.4 for $p = q$. G can be put q -purely into a constructivizable torsion-free R_q -group $\overline{G}$. Let

$$H_p = \bigoplus \{\mathbb{Z}_{p^{n+1}}^{\alpha_{pn}(A)} \mid n \in \omega\} \oplus \mathbb{Z}_{p^\infty}^{\beta_p(A)} \oplus \mathbb{Z}_p^{\gamma_p(A)},$$

$$\overline{A} = \bigoplus \{H_p \mid p \in P \setminus \{q\}\} \oplus \bigoplus \{\mathbb{Z}_{q^{n+1}}^{\alpha_{qn}(A)} \mid n \in \omega\} \oplus \mathbb{Z}_{q^\infty}^{\beta_q(A)} \oplus \overline{G}.$$

We prove that $\overline{A}$ is the required group. It is clear that the group $\overline{A}$ is constructivizable, and $A \equiv \overline{A}$. Suppose that $(\overline{A}, \nu)$ is strongly constructive,

$$m = \max\{n \mid n \in M_q\}, \quad \text{and} \quad \overline{A}_q = \{x \in \overline{A} \mid q^{m+1}x = 0\}.$$

The subgroup $\overline{A}_q$ is definable in $\overline{A}$. Therefore the factor group $\overline{A}/\overline{A}_q$ is strongly constructivizable, and its q -subgroup is divisible. The group $\overline{G}$ is contained purely in $\overline{A}/\overline{A}_q$. This contradicts Proposition 8.4. Hence $\overline{A}$ is not strongly constructivizable. The theorem is proved for case (iii).

Finally, suppose that A does not satisfy any of (i)–(iii), and (A, ν) is constructive. From Proposition 5.1, it follows that this group is strongly constructive. $\square$

9 Constructivizability of subgroups and factor groups

Let B be a subgroup of a (strongly) constructivizable group A . We consider the question: are B and A/B constructivizable? Theorems 2.2 and 2.7 give an affirmative answer for the following cases.

- (i) B is the periodic part of the constructive group (A, ν) ,
- (ii) B is a recursively enumerable subgroup of the constructive group (A, ν) such that the factor group A/B is torsion-free.

In this section, we examine this question for the following cases.

- (1) B is the reduced part of A ,
- (2) B has finite index in A ,
- (3) B is the periodic part of A .

9.1 A reduced part

In §2 and §3, we constructed (strongly) constructive p -groups such that the reduced parts of those groups were not (strongly) constructive. Here we construct a strongly constructivizable torsion-free group whose reduced part is not constructivizable. The method of proof of this result is used in §9.2 to answer a question of Macintyre. Let p be a prime number, and assume the values of the functions defined below are in the set $\{0, 1, \dots, p-1\}$. Let F be some family of recursive functions, and ν be a computable numeration of F . We construct a strongly constructive torsion-free group $(\overline{F}, \overline{\nu})$ from this pair (F, ν) in the following way. Let

$$A = \mathbb{Q}(a) \oplus \bigoplus \{\mathbb{Q}(b_{i0}) \mid i \in \omega\},$$

where $\mathbb{Q}(x) = \{rx \mid r \in \mathbb{Q}\}$, $x = a, b_{i0}$, i.e., $A \simeq \mathbb{Q}^\omega$. It is easy to verify that there is a constructivization α of A such that the set $\{a, b_{i0} \mid i \in \omega\}$ is recursive in (A, α) . Define the group $\overline{F}$ to be the subgroup of A generated by the elements $a/n, b_{ij}/n, i, j \in \omega$, such that $(p, n) = 1$, and the following condition holds for each b_{ij} .

$$b_{ij} + \nu(i)(j)a = pb_{i(j+1)}.$$

The subgroup $\overline{F}$ is recursively enumerable in (A, α) . According to Proposition 1.4, there is a numeration $\overline{\nu}$ of $\overline{F}$ such that the identity embedding $\text{id} : \overline{F} \rightarrow A$ is a recursive isomorphism of $(\overline{F}, \overline{\nu})$ into (A, α) . The group $\overline{F}$ is a torsion-free R_p -group. It is easy to check that the theory $\text{Th}(\overline{F})$ is decidable, and a Kulikov basis of $\overline{F}$ consists of a unique element a . From this fact, and Propositions 1.1 and 5.1, we obtain:

Lemma 9.1 *The pair $(\overline{F}, \overline{\nu})$ is a strongly constructive torsion-free R_p -group. The element $a \in \overline{F}$ is not divisible by p , and the factor group $\overline{F}/(a)$ is divisible.*

We need some further concepts from the theory of numerations. Let f and g be functions. We say that f is *quasi-equal* to g if

$$\exists k \forall s \geq k (f(s) = g(s)),$$

and denote this by $f \approx g$.

Let F be a family of recursive functions, and $F_{\approx} = F / \approx$. A numeration ν' of the family $F_{\approx}$ is called *computable* if there is a computable sequence of recursive functions $\{f_m \mid m \in \omega\}$ such that $f_m \in \nu'm$.

Proposition 9.2 (N. G. Khisamiev and Z. G. Khisamiev [56]). *If the reduced part R of the groups $\overline{F}$ is constructivizable, then there is a family F' of recursive functions such that $F \subseteq F'$, and $F' / \approx$ has a single-valued computable numeration.*

Proposition 9.3 (N. G. Khisamiev and Z. G. Khisamiev [56]). *There is a computable family F of recursive functions such that for every family F' of recursive functions containing F , the family $F'_{\approx}$ does not have a single-valued computable numeration.*

It follows from Propositions 9.2 and 9.3, that

Corollary 9.4 *There is a strongly constructive torsion-free abelian group whose reduced part is not constructivizable.*

Proof. Let the family F of recursive functions be defined according to Proposition 9.3, and let ν be a computable numeration of F . From the pair (F, ν) , construct the numerable group $(\overline{F}, \overline{\nu})$. By Lemma 9.1, $(\overline{F}, \overline{\nu})$ is strongly constructive. From Propositions 9.2 and 9.3, it follows that the reduced part of $\overline{F}$ is not constructivizable. $\square$

Remark 9.1 If the divisible part of a strongly constructive torsion-free abelian groups has finite rank, then the reduced part is strongly constructivizable.

Remark 9.2 The reduced part of a strongly constructive torsion-free abelian group is Π_1^0 -constructivizable.

9.2 Ordered fields

The methods used in the proof of Corollary 9.4 can be used to answer the following question of Macintyre: is the ordered field of primitive recursive real numbers constructivizable?

Recall that a real number α is called (*primitive*) *recursive* if and only if there are (primitive) recursive functions f, g, h and φ such that the following conditions are satisfied.

- (i) α is the limit of the sequence

$$\Phi(n) = (f(n) - g(n))/h(n)$$

of rational numbers,

- (ii) For each $n, n' \geq \varphi(m)$, $|\Phi(n) - \Phi(n')| < 1/2^m$.

Here the functions are total, and take only the values 0, 1. Let (S, ν) be a numerable set, and θ be an n -place relation on S . A numeration ν of S is called *positive relative to θ* if and only if the set

$$\{\langle x_1, \dots, x_n \rangle \mid \theta(x_1, \dots, x_n)\}$$

is recursively enumerable. Let F be a family of (primitive) recursive functions. For every function $f \in F$, define the (primitive) recursive real number $\alpha_f = 0, n_0 n_1 \dots$ where $n_i = f(i)$. Let $\overline{F}$ denote the ordered field of real numbers generated by $\alpha_f, f \in F$.

Proposition 9.5 *If the field $\overline{F}$ is contained in a constructivizable ordered field of real numbers, then there is a computable numeration μ of a family $S \supseteq F$ of recursive functions such that μ is positive relative to $\approx$.*

The following proposition is a minor modification of Proposition 2 from Z. G. Khisamiev [55].

Proposition 9.6 *There is a family S_0 of primitive recursive functions such that any family S of recursive functions containing S_0 does not have a computable numeration which is positive relative to $\approx$.*

An ordered field P_0 of real numbers is called *essentially non-constructivizable* if and only if P_0 is not contained in any constructivizable ordered field of real numbers.

Propositions 9.5 and 9.6 imply

Corollary 9.7 (Khisamiev [49]). *Any ordered field of real numbers containing the field of primitive recursive numbers is essentially non-constructivizable.*

Corollary 9.7 answers Macintyre's question negatively.

Corollary 9.8 *The field P of all primitive recursive real numbers is non-constructivizable.*

Indeed, it is easy to see that if $\alpha \in P$, $\alpha > 0$, then $\sqrt{\alpha} \in P$. Hence the following formula is true in P .

$$x < y \iff \exists u(y - x = u^2) \ \& \ x \neq y$$

Therefore, if the field P is constructivizable, then the ordered field

$$\langle P, +, \cdot, < \rangle$$

is constructivizable. This contradicts Corollary 9.7.

Remark 9.3 Proposition 9.5 and Corollaries 9.7 and 9.8 are true after replacing “field” by “abelian group”. Here, by abelian group we mean a subgroup of the additive group of the real numbers which contains 1.

9.3 Subgroups of finite index

The following theorem gives a sufficient condition for the constructivizability of a finite index subgroup of a constructivizable group.

Theorem 9.9 (Khisamiev [41]). *Let A be an abelian group such that the factor group A/T of A by its periodic part T has finite rank. Let B be a subgroup of A with finite index. A is (strongly) constructivizable if and only if B is.*

The following proposition shows that the condition of finiteness of rank of A/T is essential in the previous theorem.

Proposition 9.10 (Khisamiev [41]). *There is a strongly constructivizable torsion-free abelian group which has an uncountable number of non-isomorphic subgroups of index 2.*

Corollary 9.11 *There is a strongly constructivizable abelian group having a non-constructivizable subgroup of index 2.*

9.4 Periodic part

In §2, it was shown that the periodic part T and the factor group A/T of a constructivizable group A are constructivizable. For strongly constructivizable groups, the result is not true.

Proposition 9.12 (Khisamiev [39]). *There is a strongly constructivizable abelian group A such that its periodic part T is not strongly constructivizable.*

Proposition 1.1 implies:

Proposition 9.13 *The periodic part T of a strongly constructivizable abelian group A is strongly constructivizable if and only if the theory $\text{Th}(T)$ is decidable.*

Corollary 9.14 *Suppose the periodic part T of a strongly constructive abelian group A satisfies one of the following conditions.*

- (i) T is a reduced subgroup,
- (ii) T is a direct sum of a finite number of its p -components.

Then the subgroup T is strongly constructivizable.

Finally, we consider the factor group A/T .

Proposition 9.15 (Khisamiev [48]). *There is a strongly constructivizable group A with a decidable theory such that the factor group $\overline{A} = A/T$ by its periodic part T has rank 1 and is not strongly constructivizable.*

10 The arithmetic hierarchy of abelian groups

Let X be a subset of the natural numbers. In the theory of recursive functions, the classes Σ_n^X , Π_n^X and Δ_n^X , $n \in \omega$, of the arithmetic hierarchy of sets play an important role. The scheme of inclusions between these classes is as follows.

$$\Delta_0^X = \Pi_0^X = \Sigma_0^X = \Delta_1^X = \Sigma_1^X \cap \Pi_1^X \subsetneq \Sigma_1^X \subsetneq \Delta_2^X = \Sigma_2^X \cap \Pi_2^X \subsetneq \Sigma_2^X \subsetneq \dots \quad (*)$$

$$\qquad \qquad \qquad \subsetneq \Pi_1^X \subsetneq \qquad \qquad \qquad \subsetneq \Pi_2^X \subsetneq$$

Let Y denote one of these classes. If K is a class of groups, then let $K(Y)$ denote the class of Y -constructivizable groups in K . If we replace Y by $K(Y)$ in $(*)$, then we obtain the arithmetic hierarchy of groups for the class K . This raises the question: which inclusions from $(*)$ are strong? In this section, the question is answered for the following classes.

- (1) Torsion-free abelian groups.
- (2) Direct sums of cyclic groups.
- (3) Direct sums of cyclic and quasicyclic p -groups.

10.1 Torsion-free groups

From Corollary 2.12 we obtain:

Corollary 10.1 *If a torsion-free abelian group is Σ_n^X -constructivizable, then it is Δ_n^X -constructivizable.*

Let A be a torsion-free abelian group of countable rank, and fix a basis $\{\alpha_i \mid i \in \omega\}$ for it. The characteristic $\chi(A)$ of A is the set

$$\chi(A) = \left\{ (\alpha_0, \dots, \alpha_{n-1}, p, r) \mid r > 0, p \in P, \alpha_i \in \mathbb{Z}, \right.$$

$$\text{g.c.d.}(\alpha_0, \dots, \alpha_{n-1}, p) = 1,$$

$$\left. A \models \exists y(p^r y = \alpha_0 a_0 + \dots + \alpha_{n-1} a_{n-1}), n \in \omega \right\}.$$

Proposition 10.2 *If the characteristic $\chi(A)$ of a group A belongs to the class Σ_{n+1}^X , $n \geq 1$, then there is a numeration μ of A such that (A, μ) is a Π_n^X -constructivizable group, and (A, μ) has a recursively enumerable basis.*

From Theorem 2.2, Corollary 10.1 and Proposition 10.2, we obtain:

Theorem 10.3 (Khisamiev [51]). *If a torsion-free group is Σ_{n+1}^X -constructivizable, $n \in \omega$, then it is Π_n^X -constructivizable.*

Corollary 10.4 *Any class $L(Y)$ of the arithmetic hierarchy for the class L of torsion-free abelian groups coincides with one of the classes $L(\Pi_n^X)$, $n \in \omega$. For each n there is an abelian group $A_n \in L(\Pi_{n+1}^X) \setminus L(\Pi_n^X)$.*

Proof. The first part of the corollary follows from Theorem 10.3. We prove the second part. If $n = 0$, then the existence of a group A_1 follows from Corollary 9.4 and Remark 9.2. Let $n \geq 1$ and S_n be a set of prime numbers such that $S_n \in \Sigma_{n+2}^0 \setminus \Sigma_{n+1}^0$. Let A_n denote the subgroup of the additive group of rational numbers generated by the set $\{1/p \mid p \in S_n\}$. By Proposition 10.2, A_n is Π_{n+1}^0 -constructivizable. If A_n is Π_n^0 -constructivizable, then we have $S_n \in \Sigma_{n+1}^0$. This contradicts the choice of S_n . $\square$

10.2 Periodic abelian groups

The following propositions give criteria for Σ_1^X - and Π_1^X -constructivizability of groups. These propositions are also of independent interest.

Let $\{q_i \mid i \in \omega\}$ be a sequence of prime numbers, and $\{n_i \mid i \in \omega\}$ be a sequence of natural numbers. We let $A = \bigoplus \{\mathbb{Z}_{q_i^{n_i}} \mid i \in \omega\}$. The set

$$\chi(A) = \{(p^n, k) \mid n, k > 0, \exists i_1, \dots, i_k (q_{i_1}^{n_{i_1}} = \dots = q_{i_k}^{n_{i_k}} = p^n)\}$$

is called the characteristic of A .

Proposition 10.5 (Khisamiev [51]). *A is Σ_1^X -constructivizable if and only if the characteristic $\chi(A)$ is a Σ_2^X -set.*

Let the function $g(x)$ be defined everywhere. The least natural number m , which is a value of the function in infinitely many places, is called a lower limit of g , and denoted by $\underline{\lim} g(x) = m$. If there is no such number, then $\underline{\lim} g(x) = \infty$.

Proposition 10.6 (Khisamiev [51]). *The abelian p -group*

$$G = \bigoplus \{\mathbb{Z}_{p^{\alpha_i}} \mid i \in \omega\},$$

$\alpha_i \in \omega \cup \{\infty\}$, *is Π_1^X -constructivizable if and only if there is an X -recursive function $f(i, x)$ such that*

$$G \simeq \bigoplus \{\mathbb{Z}_{p^{\beta_i}} \mid i \in \omega\}$$

where $\beta_i = \lim_{\lambda} \lambda x f(i, x)$.

Using Theorem 3.4 and Propositions 10.5 and 10.6, we can prove:

Theorem 10.7 (Khisamiev [51]). *Let L_0 be the class of direct sums of cyclic groups. All classes of the arithmetic hierarchy for L_0 are different.*

Theorem 10.8 (Khisamiev [51]). *Let L_p be the class of groups which are direct sums of cyclic and quasicyclic groups. The following relations hold for the classes of the arithmetic hierarchy of L_p .*

- (i) $L_p(\Delta_{n+1}^X) = L_p(\Pi_n^X)$,
- (ii) $L_p(\Sigma_n^X) \subsetneq L_p(\Pi_n^X)$.

References

- [1] K. T. Abdrazakov and N. G. Khisamiev, A criterion for strong constructivizability of a class of abelian p -groups (Russian), *Sibirsk. Math. Zh.*, **25** (1984) 3–8; [translated in: *Siberian Math. J.*, **25** (1984) 511–515].
- [2] C. J. Ash, J. F. Knight and S. E. Oates, Recursive reduced Abelian p -groups of small length, *Logic paper*, (preprint), **74** (1991) 1–28.
- [3] V. Ya. Blotchitsyn and K. Kh. Zakir'yanov, Constructive abelian groups (Russian), *Algebra i Teor. Chisel*, Alma-Ata, (1978) 18–25.
- [4] R. Deissler, Minimal and prime models of complete theories of torsion free abelian groups, *Algebra Universalis*, **9** (1979) 250–265.

- [5] V. P. Dobritsa, Computability of certain classes of constructive algebras (Russian), *Sibirsk. Math. Zh.*, **18** (1977) 570–579; [translated in: *Siberian Math. J.*, **18** (1977) 406–413].
- [6] V. P. Dobritsa, On constructivizable abelian groups (Russian), *Sibirsk. Math. Zh.*, **22** (1981) 208–213, 239.
- [7] V. P. Dobritsa, Some constructivizations of abelian groups (Russian), *Sibirsk. Math. Zh.*, **24** (1983) 18–25; [translated in: **24** (1983) 167–173].
- [8] V. P. Dobritsa and S. S. Goncharov, An example of a constructive abelian group with a non-constructivizable reduced subgroup (Russian), in: *Proc. 4th. All-Union Conf. Math. Logic, Kishinev, (1976)* 33.
- [9] V. P. Dobritsa, N. G. Khisamiev and A. T. Nurtazin, Constructive periodic abelian groups (Russian), *Sibirsk. Math. Zh.*, **19** (1978) 1260–1265; [translated in: *Siberian Math. J.*, **19** (1978) 886–890].
- [10] R. G. Downey and S. A. Kurtz, Recursion theory and ordered groups, *Ann. Pure Appl. Logic*, **32** (1986) 137–151.
- [11] B. N. Drobotun, On the algorithmical dimension of nilpotent groups (Russian), 7th. *Kazakh. Mezhvuzov. Konf. Mat. i Mekh.*, Karaganda, (1981) 133–134.
- [12] B. N. Drobotun, On autostability of nilpotent groups (Russian), in: *Proc. 16th. All-Union Conf. Algebra, Leningrad, (1981)* 48–49.
- [13] V. D. Dzgoev, Constructivizations of direct products of algebraic systems (Russian), *Algebra i Logika*, **21** (1982) 138–148; [translated in: *Algebra and Logic*, **21** (1982) 88–96].
- [14] P. C. Eklof and E. R. Fischer, The elementary theory of abelian groups, *Ann. Math. Logic*, **4** (1972) 115–171.
- [15] Yu. L. Ershov, Existence of constructivizations (Russian), *Dokl. Akad. Nauk SSSR*, **204** (1972) 1041–1044 [translated in: *Soviet Math. – Dokl.*, **13** (1972) 779–783].
- [16] Yu. L. Ershov, Constructive models (Russian), in: *Izbr. Vopr. Alg. i Log.*, [Selected Questions in Algebra and Logic], *Sbornik posvjascen. pamjati A. I. Mal'ceva*, [A collection dedicated to the memory of A. I. Mal'tsev], *Izdat. Nauk Sibirsk. Otdel.*, Novosibirsk, (1973) 111–130.

- [17] Yu. L. Ershov, Skolem functions and constructive models (Russian), *Algebra i Logika*, **12** (1973) 644–654, 735; [translated in: *Algebra and Logic*, **12** (1973) 368–373].
- [18] Yu. L. Ershov, *Theory of Numerations* (Russian), *Monographs in Math. Logic and Foundations of Math.*, (Nauka, Moskva, 1977).
- [19] Yu. L. Ershov, *Decision Problems and Constructivizable Models* (Russian), (*Mathematical Logic and Foundations of Mathematics*, Nauka, Moskva, 1980).
- [20] Yu. L. Ershov and E. A. Palyutin, *Mathematical Logic* (Russian), 2nd. edn., (Nauka, Moskva, 1987). [English translation of 1st. edn., translated by V. Shokurov, (MIR Publishers, Moscow, 1984)].
- [21] L. J. Feiner, Hierarchies of Boolean algebras, *J. Symbolic Logic*, **35** (1970) 365–374.
- [22] A. Fröhlich and J. C. Shepherdson, Effective procedures in field theory, *Philos. Trans. Roy. Soc. London, Ser. A*, **248** (1956) 407–432.
- [23] L. Fuchs, Infinite abelian groups, Vol. 1, *Pure Appl. Math.*, **36** (1970).
- [24] L. Fuchs, Infinite abelian groups, Vol. 2, *Pure Appl. Math.*, **36–II** (1973).
- [25] S. S. Goncharov, The constructivizability of superatomic Boolean algebras (Russian), *Algebra i Logika*, **12** (1973) 31–40, 120; [translated in: *Algebra and Logic*, **12** (1973) 17–22].
- [26] S. S. Goncharov, Certain properties of the constructivization of Boolean algebras (Russian), *Sibirsk. Math. Zh.*, **16** (1975) 264–278; [translated in: *Siberian Math. J.*, **16** (1975) 203–214].
- [27] S. S. Goncharov, Strong constructivizability of homogeneous models (Russian), *Algebra i Logika*, **17** (1978) 363–388, 490; [translated in: *Algebra and Logic*, **17** (1978) 247–263].
- [28] S. S. Goncharov, Autostability of models and abelian groups (Russian), *Algebra i Logika*, **19** (1980) 23–44; [translated in: *Algebra and Logic*, **19** (1980) 13–27].
- [29] S. S. Goncharov, Totally transcendental theory with a nonconstructivizable prime model (Russian), *Sibirsk. Math. Zh.*, **21** (1980) 44–51; [translated in: *Siberian Math. J.*, **21** (1980) 32–37].

- [30] S. S. Goncharov, Groups with a finite number of constructivizations (Russian), *Dokl. Akad. Nauk SSSR*, **256** (1981) 269–272; [translated in: *Soviet Math. – Dokl.*, **23** (1981) 58–61].
- [31] S. S. Goncharov, Countable Boolean Algebras (Russian), (Nauka. Sibirsk. Otdel., Novosibirsk, 1988).
- [32] S. S. Goncharov and B. N. Drobotun, On the algorithmic dimension of nilpotent groups (Russian), *Sibirsk. Math. Zh.*, **30** (1989) 52–60, 225; [translated in: *Siberian Math. J.*, **30** (1989) 210–217].
- [33] S. S. Goncharov, A. V. Molokov and N. S. Romanovskii, Nilpotent groups of finite algorithmic dimension, *Sibirsk. Math. Zh.*, **30** (1989) 82–88; [translated in: *Siberian Math. J.*, **30** (1989) 63–68].
- [34] S. S. Goncharov and A. T. Nurtazin, Constructive models of complete decidable theories (Russian), *Algebra i Logika*, **12** (1973) 125–142, 243; [translated in: *Algebra and Logic*, **12** (1973) 67–77].
- [35] L. Harrington, Recursively presentable prime models, *J. Symbolic Logic*, **39** (1974) 305–309.
- [36] M. J. Hassett, Recursive equivalence types and groups, *J. Symbolic Logic*, **34** (1969) 13–20.
- [37] M. I. Kargapolov and Yu. I. Merzljakov, Fundamentals of the Theory of Groups, translation of *Osnovy teorii grupp*, 2nd. edn., 1977, translated by R. G. Burns, *Grad. Texts in Math.*, **62** (1979); [4th. edn., *Fismatlit. Nauka*, Moscow, 1996].
- [38] N. G. Khisamiev, Strongly constructive models of a decidable theory (Russian), *Izv. Akad. Nauk Kazakh. SSR, Ser. Fiz.-Mat.*, **1** (1974) 83–84.
- [39] N. G. Khisamiev, The periodic part of a strongly constructivizable abelian group (Russian), *Teor. i Priklad. Zad. Mat. i Mekh.*, Alma-Ata, Nauka Kazakh. SSR, (1977) 299–303, 318.
- [40] N. G. Khisamiev, Strongly constructive periodic abelian groups (Russian), *Izv. Akad. Nauk Kazakh. SSR Ser. Fiz.-Mat.*, (1978) 58–62, 92.
- [41] N. G. Khisamiev, Subgroups of finite index of abelian groups (Russian), *Izv. Akad. Nauk Kazakh. SSR Ser. Fiz.-Mat.*, (1979) 43–47, 89.

- [42] N. G. Khisamiev, Constructivizable ordinals are strongly constructivizable (Russian), *Sibirsk. Math. Zh.*, **21**:1 (1980) 231 (abstract).
- [43] N. G. Khisamiev, Criterion for constructivizability of a direct sum of cyclic p -groups (Russian), *Izv. Akad. Nauk Kazakh. SSR Ser. Fiz.-Mat.*, (1981) 51–55, 86.
- [44] N. G. Khisamiev, Strongly constructive abelian p -groups (Russian), *Algebra i Logika*, **22** (1983) 198–217; [translated in: *Algebra and Logic*, **22** (1983) 142–158].
- [45] N. G. Khisamiev, Connection between constructivizability and strong constructivizability for different classes of abelian groups (Russian), *Algebra i Logika*, **23** (1984) 319–335; [translated in: *Algebra and Logic*, **23** (1984) 220–233].
- [46] N. G. Khisamiev, Non-strong constructivizability of a factor group of an abelian group by its periodic part (Russian), in: *Proc. 8th. Kazakh. Mezhvuzov. Konf. Mat. i Mekh.*, part 1, *Mat.*, Alma-Ata, (1984) 202.
- [47] N. G. Khisamiev, Theory of abelian groups with constructive models (Russian), *Sibirsk. Math. Zh.*, **27** (1986) 128–143, 215; [translated in: *Siberian Math. J.*, **27** (1986) 572–585].
- [48] N. G. Khisamiev, Hierarchies of torsion-free abelian groups (Russian), *Algebra i Logika*, **25** (1986) 205–226, 244; [translated in: *Algebra and Logic*, **25** (1986) 128–142].
- [49] N. G. Khisamiev, Non constructivizability of certain ordered fields of real numbers (Russian), *Sibirsk. Math. Zh.*, **28** (1987) 193–195; [translated in: *Siberian Math. J.*, **28** (1987) 845–847].
- [50] N. G. Khisamiev, On constructive abelian p -groups (Russian), in: *Proc. 19th. All-Union Conf. Algebra*, part 2, L'vov, (1987) 299.
- [51] N. G. Khisamiev, The arithmetic hierarchy of abelian groups (Russian), *Sibirsk. Math. Zh.*, **29** (1988) 144–159; [translated in: *Siberian Math. J.*, **29** (1988) 987–999].
- [52] N. G. Khisamiev, A criterion for the constructivizability of torsion-free abelian] groups (Russian), in: *Proc. 9th. All-Union Conf. Math. Logic*, Leningrad, (1988) 168.

- [53] N. G. Khisamiev, Constructive reduced abelian p -groups (Russian), in: Proc. Internatl. Conf. Algebra in memory of Mal'tsev (1909–1967), Novosibirsk, (1989) 147.
- [54] N. G. Khisamiev, Constructive abelian p -groups, *Siberian Adv. Math.*, **2** (1992) 68–113.
- [55] Z. G. Khisamiev, Calculable enumerations and equivalence relations (Russian), *Sibirsk. Math. Zh.*, **27** (1986) 182–187, 207; [translated in: *Siberian Math. J.*, **27** (1986) 779–783].
- [56] Z. G. Khisamiev and N. G. Khisamiev, Nonconstructivizability of the reduced part of a strongly constructive torsion-free abelian group (Russian), *Algebra i Logika*, **24** (1985) 108–118, 123; [translated in: *Algebra and Logic*, **24** (1985) 69–76].
- [57] L. Ya. Kulikov, On the theory of abelian groups of arbitrary power (Russian), *Mat. Sb. (N.S.)*, **16** (1945) 129–160; [English summary: *ibid.*, 161–162].
- [58] A. G. Kurosh, Primitive torsionsfreie abelsche Gruppen vom endlichen Range, [Primitive torsion-free abelian groups of finite rank], *Ann. of Math.*, Ser. 2, **38** (1937) 175–203.
- [59] A. H. Lachlan and E. W. Madison, Computable fields and arithmetically definable ordered fields, *Proc. Amer. Math. Soc.*, **24** (1970) 803–807.
- [60] I. V. Latkin, Algorithmic complexity of the problem of occurrence in commutants and members of the lower central series (Russian), *Sibirsk. Math. Zh.*, **28** (1987) 102–110; [translated in: *Siberian Math. J.*, **28** (1987) 772–779].
- [61] I. V. Latkin, Hierarchies of torsion-free nilpotent groups (Russian), in: Proc. 9th. All-Union Conf. Math. Logic, Leningrad, (1988) 91.
- [62] M. Lerman and J. H. Schmerl, Theories with recursive models, *J. Symbolic Logic*, **44** (1979) 59–76.
- [63] C. R. Lin, The effective content of Ulm's Theorem, in: *Aspects of Effective Algebra*, (Proc. Conf. Monash Univ., Clayton, Australia, Aug. 1–4, 1979), J. N. Crossley (ed.), (Upside Down A Book Co., Yarra Glen, Victoria, Australia, 1981) 147–160.

- [64] C. R. Lin, Recursively presented abelian groups: effective p -group theory I, *J. Symbolic Logic*, **46** (1981) 617–624.
- [65] A. I. Mal'tsev, Torsion-free abelian groups of finite rank (Russian), *Mat. Sb. (N.S.)*, **4** (1938) 45–67; [German summary: *ibid.*, 67–68].
- [66] A. I. Mal'tsev, Constructive algebras I (Russian), *Uspekhi Mat. Nauk*, **16** (1961) 3–60; [translated in: *Constructive algebras I*, *Russian Math. Surveys*, **16**:3 (1961) 77–129; also in: *The Metamathematics of Algebraic Systems*, *Collected Papers: 1936–1967*, translated and edited by B. F. Wells III, *Stud. Logic Found. Math.*, **66** (1971), Ch. 18, 148–200].
- [67] A. I. Mal'tsev, On recursive abelian groups (Russian), *Dokl. Akad. Nauk SSSR*, **146** (1962) 1009–1012; [translated in: *Soviet Math. – Dokl.*, **3** (1962) 1431–1434; also in: *The Metamathematics of Algebraic Systems*, *Collected Papers: 1936–1967*, translated and edited by B. F. Wells III, *Stud. Logic Found. Math.*, **66** (1971), Ch. 24, 282–286].
- [68] A. I. Mal'tsev, *Algorithms and Recursive Functions* (Russian), (Izdat. Nauka, Moscow, 1965); [translated: by L. F. Boron, L. E. Sanchis, J. Stillwell and K. Iseki, (Wolters-Noordhoff Publishing, Groningen, 1970)].
- [69] J. Mead, Recursive prime models for Boolean algebras, *Colloq. Math.*, **41** (1979) 25–33.
- [70] K. A. Meirembekov, Preservation of stability under finite extension of a group (Russian), *Algebra i Logika*, **25** (1986) 273–291; [translated in: *Algebra and Logic*, **25** (1986) 171–183].
- [71] A. V. Molokov, Prime models of theories of abelian groups (Russian), in: *Proc. 6th. All-Union Conf. Math. Logic*, Tbilisi, (1982) 116.
- [72] A. V. Molokov, Prime models of theories of abelian groups (Russian), *Nekotorye Problemy i Zad. Algebr i Analiza*, Novosibirsk. Gos. Univ., Novosibirsk, (1985) 113–119.
- [73] A. S. Morozov, Countable homogeneous Boolean algebras (Russian), *Algebra i Logika*, **21** (1982) 269–282; [translated in: *Algebra and Logic*, **21** (1982) 181–190].
- [74] A. S. Morozov, Strong constructivizability of countable saturated Boolean algebras (Russian), *Algebra i Logika*, **21** (1982) 193–203; [translated in: *Algebra and Logic*, **21** (1982) 130–137].

- [75] P. S. Novikov, On the algorithmic unsolvability of the problem of identity (Russian), *Dokl. Akad. Nauk SSSR*, **85** (1952) 709–712.
- [76] A. T. Nurtazin Computable classes and algebraic criteria of autostability, summary of scientific thesis (Russian), (Math. Inst. Siberian Branch of SSSR Acad. Sci., Novosibirsk, 1974).
- [77] A. T. Nurtazin, On constructive groups (Russian), in: *Proc. 4th. All-Union Conf. Math. Logic*, (Kishinev, 1976), (1976) 106.
- [78] M. G. Peretyat'kin, Strongly constructive models and enumerations of the Boolean algebra of recursive sets (Russian), *Algebra i Logika*, **10** (1971) 535–557; [translated in: *Algebra and Logic*, **10** (1971) 332–345].
- [79] M. G. Peretyat'kin, Every recursively enumerable extension of a theory of linear order has a constructive model (Russian), *Algebra i Logika*, **12** (1973) 211–219, 244; [translated in: *Algebra and Logic*, **12** (1973) 120–124].
- [80] M. G. Peretyat'kin, A criterion for strong constructivizability of a homogeneous model (Russian), *Algebra i Logika*, **17** (1978) 436–454, 491; [translated in: *Algebra and Logic*, **17** (1978) 290–301].
- [81] M. O. Rabin, Computable algebra, general theory and theory of computable fields, *Trans. Amer. Math. Soc.*, **95** (1960) 341–360.
- [82] F. Richman, The constructive theory of countable abelian p -groups, *Pacific J. Math.*, **45** (1973) 621–637.
- [83] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, (1st. edn., McGraw-Hill, New York-Toronto, Ont.-London, 1967; 2nd. edn., MIT Press, Cambridge, Mass., London, 1987).
- [84] R. L. Smith, Effective group structures on Π_1^0 classes of functions, *Notices Amer. Math. Soc.*, **25** (1978) A-384 (abstract).
- [85] R. L. Smith, Abelian recursively profinite groups, *Notices Amer. Math. Soc.*, **25** (1978) A-507 (abstract).
- [86] R. L. Smith, The theory of profinite groups with effective presentation, *Dissert. Abstr. Intern. B*, **40** (1979) 1758B–1759B.
- [87] R. L. Smith, Two theorems on autostability in p -groups, in: *Logic Year 1979-1980*, (Proc. Seminars and Conf. Math. Logic, Univ. Connecticut, Storrs, CT, 1979/80), M. Lerman, J. H. Schmerl and R. I. Soare, (eds.), *Lecture Notes in Math.*, **859** (1981) 302–311.

- [88] E. Specker, Nicht konstruktiv beweisbare Sätze der Analysis, *J. Symbolic Logic*, **14** (1949) 145–158.
- [89] W. Szmielew, Elementary properties of Abelian groups, *Fund. Math.*, **41** (1955) 203–271.
- [90] V. A. Uspenskii, Lectures on computable functions (Russian), Gosudarstv. Izdat. Fiz.-Mat. Lit., (Moscow, 1960).

N. G. Khisamiev
East-Kazakhstan Technical University,
Ust-Kamenogorsk,
Kazakhstan.
E-mail: zoot@wktu.ksisti.alma-ata.su

This Page Intentionally Left Blank

Chapter 18

Recursive and On-Line Graph Coloring

H. A. Kierstead*

Contents

- Introduction
- 1. Highly recursive graphs
- 2. Recursive ordered sets
- 3. On-line coloring
- 4. On-line χ -bounded classes
- 5. An application: Dynamic Storage Allocation
- References

1 Introduction

In this article we survey results of the last twenty years concerning recursive and on-line coloring of graphs. The subject began in the seventies as part of the effort by logicians to understand the effective content of various existence theorems in mathematics. One of the areas chosen for investigation was graph theory, and in particular, graph coloring. It is well known, and easy to prove using any of a number of forms of the compactness theorem, that for any finite number k , an infinite graph G is k -colorable iff every finite subgraph of G is k -colorable. It also is not hard to show that this is not an effective result. There exists a recursive k -colorable graph G which cannot be k -colored by any recursive function. Given such a state of affairs one can either study the degrees of unsolvability of k -colorings of G or try to find recursive colorings of G which do not use too many extra colors. Here we take the second course. Our goal is to discover interesting classes of recursive k -colorable graphs which can be recursively colored with a bounded number of colors.

*Research partially supported by Office of Naval Research grant N00014-90-J-1206.

It turns out that in most cases simply being recursive is not enough. For instance, there are acyclic (2-colorable) recursive graphs which cannot be recursively colored using finitely many colors. Some additional local information is required. The first interesting results were about highly recursive graphs. In highly recursive graphs every vertex has only finitely many neighbors and these neighbors can be effectively calculated. These results are discussed in Section 2. The problem of covering an ordered set by chains or antichains is closely related to graph coloring. However local information is not as vital. We shall see that if a recursive ordered set can be covered by finitely many chains or antichains, then it can be recursively covered by finitely many chains or antichains. These results are presented in Section 3. They give rise to more general questions concerning recursive colorings of recursive graphs which are answered by a very general theorem in Section 5.

Problems concerning recursive coloring of recursive graphs are very closely related to the problem from computer science of on-line graph coloring. In this problem the vertices of a (usually finite) graph are presented one at a time and an *on-line* algorithm is required to assign them a color based only on information about edges of the graph between vertices already presented and the colors previously assigned to these vertices. Although there is no general theorem, results about on-line coloring can be routinely converted to results about recursive coloring of recursive (but not highly recursive) graphs and vice versa via diagonal arguments. But the study of on-line algorithms seems to be broader. There are many questions and results about on-line coloring which do not arise naturally from the study of recursive coloring. On-line coloring is introduced in Section 4. In the remaining sections all results are presented in terms of on-line coloring; most have easy translations to recursive coloring. In the last section we discuss a surprising application of an on-line coloring algorithm for the design of a polynomial time approximation algorithm for an NP-complete storage problem. In the remainder of this section we review our notation and terminology.

A *graph* is an ordered pair $G = (V, E)$ where E is a set of two element subsets of V . The elements of V are called vertices, and the elements of E are called edges. The sets of vertices and edges may be denoted by $V(G)$ and $E(G)$, respectively. Two vertices x and y are said to be adjacent if $\{x, y\} \in E$. In this case we may write $x \sim y$ or $xy \in E$. If x and y are not adjacent, they are *non-adjacent* or *independent*. A set of pairwise non-adjacent vertices is called an *independent* set. The *neighborhood* of a subset X of V is $N(X) = \{y \in V : x \sim y, \text{ for some } x \in X\}$. If $X = \{x\}$, we

abuse notation by writing $N(x)$. The *degree* of a vertex x is $\delta(x) = |N(x)|$. The degree of a graph is the maximum $\Delta(G)$ of the degrees of the vertices of the graph. Let $N^0(X) = X$ and $N^{i+1}(X) = N(N^i(X))$. A k -*coloring* c of a graph G is a function $c : V \rightarrow C$ such that $|C| = k$ and if $x \sim y$ then $c(x) \neq c(y)$. In this case we say that G is k -*colorable*. The *chromatic number* of a graph G , denoted $\chi(G)$, is the least integer k such that G is k -colorable. The elements of C are called colors, and, for any $\alpha \in C$, the set $\{v \in V : c(v) = \alpha\}$ is called a color class.

A graph $G = (V, E)$ is *recursive* if both V and E are recursive sets. A k -coloring f of G is *recursive* if f is a recursive function. In this case G is said to be *recursively k -colorable*. The *recursive chromatic number* of G , denoted $\chi_{\text{rec}}(G)$ is the least k such that G is recursively k -colorable. If Γ is a class of graphs, $\chi_{\text{rec}}(\Gamma)$ denotes the least k such that each graph G in Γ is recursively k -colorable.

A *path* is a set of vertices $\{v_1 \sim v_2 \sim \dots \sim v_t\}$ with *end points* v_1 and v_t . Two vertices of a graph are *connected* if they are the end points of a path. Connectedness is an equivalence relation. The equivalence classes of this relation are called connected components. A graph is connected if it has only one connected component. A *cycle* is a set of vertices $\{v_1 \sim v_2 \sim \dots \sim v_t \sim v_1\}$. A graph is *acyclic* if it does not contain a cycle. A *tree* is an acyclic connected graph. A *clique* in a graph $G = (V, E)$ is a set of vertices K such that any two vertices are adjacent. The *clique number* of G , denoted $\omega(G)$, is the size of the largest clique in G . The clique number of G is an obvious lower bound on the chromatic number of G . The size of the largest independent set in a graph G is denoted by $\alpha(G)$.

A graph $H = (W, F)$ is an *induced subgraph* of G if both $W \subset V$ and $xy \in F$ iff $xy \in E$, for all x and y in W . In this case we say that W *induces* H in G and write $H = G[W]$.

2 Highly recursive graphs

During the seventies and early eighties, several authors, including Manaster and Rosenstein [31], Bean [1], Schmerl [35, 36], and Kierstead [16] studied the relationship between the chromatic number and the recursive chromatic number of various classes of graphs. The subject nearly never got off the ground because of the following negative result of Bean. Note that acyclic graphs are certainly 2-colorable.

Theorem 2.1 (Bean [1]) *There exists an acyclic recursive graph which cannot be recursively colored with finitely many colors.*

This led Bean to strengthen the notion of a recursive graph. A graph $G = (V, E)$ is *highly recursive* if G is recursive, $\delta(x)$ is finite for every x , and δ is a recursive function. The advantage of working with highly recursive graphs is that the neighborhood of any finite set S of vertices is finite, and moreover, an algorithm can compute this neighborhood by searching through the vertices of G until $\delta(v)$ neighbors of v have been found for every $v \in S$. Thus an algorithm can examine $N(v)$, $N(N(v))$, $\dots$, $N^i(v)$ before committing itself to a color for v . Improving an earlier result of Bean [1], Schmerl proved the following fundamental theorem for highly recursive graphs.

Theorem 2.2 (Schmerl [35]) *Every k -colorable highly recursive graph is recursively $(2k - 1)$ -colorable. Moreover, there exist k -colorable, highly recursive graphs which are not recursively $(2k - 2)$ -colorable, for every $k \geq 2$.*

Proof. Let $G = (V, E)$ be a k -colorable highly recursive graph. Then the vertices of G are natural numbers, say $\{v_1 \ll v_2 \ll \dots\}$, where $\ll$ is (non-standard) notation for the standard order on the natural numbers. We must exhibit an algorithm for $(2k - 1)$ -coloring G . First note that it is easy to recursively $2k$ -color G . Use two disjoint k -sets $A = \{\alpha_1, \dots, \alpha_k\}$ and $B = \{\beta_1, \dots, \beta_k\}$ of colors. First color $V_1 = \{v_1\}$ with α_1 . Then color the finite set $V_2 = (N(V_1) \cup \{v_2\}) - V_1$ with colors from B by trial and error. Next color $V_3 = (N(V_2) \cup \{v_3\}) - V_2$ with colors from A , etc..

In order to save a color, let $A \cap B = \{\alpha_1\} = \{\beta_1\}$. Recursively partition V into sets $V_1, V_2, \dots$ by setting $V_1 = \{v_1\}$ and $V_{i+1} = (N^2(V_i) \cup \{v_{i+1}\}) - U_i$, where $U_i = V_1 \cup \dots \cup V_i$. Let $X_{i+1} = N(U_i) - U_i$ and $Y_{i+1} = V_{i+1} - X_{i+1}$. Suppose we have colored U_i with $A \cup B$ so that:

- (i,a) Colors from only one set, say A , are used on vertices of Y_i , and
- (i,b) this coloring can be extended to a coloring f of $U_i \cup X_{i+1}$ so that only colors for A are used on X_{i+1} .

It suffices to construct a coloring h of U_{i+1} such that $(i + 1, a)$ and $(i + 1, b)$ hold. By trial and error find a coloring g of $V_{i+1} \cup X_{i+2}$ which only uses colors from B . Define h by

$$\begin{aligned} h(v) &= f(v), & \text{if } v \in U_i, & \text{ or both } v \in X_{i+1} \text{ and } f(v) \neq \alpha_1; \\ h(v) &= g(v), & \text{otherwise.} \end{aligned}$$

It is easy to check that h is a (proper) coloring of U_{i+1} which satisfies $(i+1, a)$ and $(i+1, b)$.

Next we show that there is a k -colorable highly recursive graph which cannot be recursively $(2k-2)$ -colored. We use a standard diagonal argument to construct G . The vertex set of G will consist of disjoint parts $V_1, V_2, \dots$, one for each of the countably many possibilities $\varphi_1, \varphi_2, \dots$ for recursively coloring G . There will be no adjacencies between vertices in different parts. Thus, to ensure that G is k -colorable it suffices to ensure that each part is k -colorable. The e -th part will be constructed using a “wait and see” argument so that φ_e does not properly $(2k-2)$ -color V_e . We construct a k -colorable graph G_e on V_e by presenting the vertices of V_e one at time so that both the adjacency relations between a newly presented vertex v and previously presented vertices, and the degree of v , are determined at the time v is presented.

In the case $k = 2$, the construction of G_e is very simple. We begin by building two disjoint paths $(x_1, \dots, x_i)$ and $(y_1, \dots, y_i)$. Each time a new vertex x_j or y_j is presented, we let its degree be two if $j > 1$ and one if $j = 1$. If φ_e eventually uses the same color on x_1 and y_1 , then the only way for φ_e to 2-color G_e is to give the same color to x_i and y_i . At this time we complete the construction of G_e by presenting two more vertices x and y so that $x_i \sim x \sim y \sim y_i$. Then φ_e will be forced to use three colors on the set $\{x_i, x, y, y_i\}$. If φ_e eventually uses different colors on x_1 and y_1 , φ_e must use different colors on x_i and y_i . Thus we complete the construction by presenting one more vertex x such that $x_i \sim x \sim y_i$. Then φ_e will be forced to use three colors on the set $\{x_i, x, y_i\}$. If φ_e never colors one of x_1 and y_1 , then φ_e is not a 2-coloring of G_e , which will consist of two one-way infinite paths.

For the general argument we replace vertices of the paths by copies of a special subgraph M and the edges of the path by *connection sets* of edges between the copies of M . The subgraph M will be k -colorable as will be a path consisting of copies of M connected by connection sets. A $(2k-2)$ -coloring of M will induce a parity and two copies of M connected by a connection set will have opposite parity.

Let $M = (V, E)$ where $V = \{m_{i,j} : 1 \leq i, j \leq k\}$ and $m_{ij} \sim m_{st}$ iff both $i \neq s$ and $j \neq t$. We visualize M as a matrix and refer to the i -th row $R_i = \{m_{ij} : 1 \leq j \leq k\}$ and the j -th column $C_j = \{m_{ij} : 1 \leq i \leq k\}$. Note

that the diagonal $\{m_{ii} : 1 \leq i \leq k\}$ of M is a k -clique and that the rows and columns of M are independent sets. Thus we can k -color M either by using the rows or the columns as color classes. If c is a coloring of M we say that M is c -row (column) colorful if c uses k different colors on some row (column) of M .

Claim 1 *If c is a $(2k - 2)$ coloring of M , then M is either c -row or c -column colorful.*

Proof. Otherwise each row R_i has a repeated color ρ_i and each column C_j has a repeated color γ_j . It is easy to check that since c is a coloring, $|\{\rho_s, \gamma_s : 1 \leq s \leq k\}| = 2k$. But this contradicts c being a $(2k - 2)$ -coloring. $\square$

Claim 2 *If c is a $(2k - 2)$ -coloring of M , then M is not both c -row and c -column colorful.*

Proof. Suppose row R_i witnesses that M is c -row colorful and column C_j witnesses that M is c -column colorful. Then it is easy to check that c must use $2k - 1$ colors on $R_i \cup C_j$. But this contradicts c being a $(2k - 2)$ -coloring. $\square$

We say that two copies of M which have been colored by c have the same parity if they are both c -row colorful or both c -column colorful. If one is c -row colorful and the other is c -column colorful, then we say that they have opposite parity. Claims 1 and 2 show that these notions are well defined when c is a $(2k - 2)$ -coloring. Next we define connection sets so that two copies of M joined by a connection set have opposite parity.

Let A and B be two copies of M where each m_{ij} in M corresponds to a_{ij} in A and b_{ij} in B . We shall say that A and B are adjacent in G_e if A and B are both induced subgraphs of G_e and $a_{ij} \sim b_{st}$ iff both $i \neq t$ and $j \neq s$. The edges $\{\{a_{ij}, b_{st}\} : i \neq t \text{ and } j \neq s\}$ are the connection sets referred to above. The following claim is proved in much the same way as the previous claim.

Claim 3 *If A is adjacent to B in G_e and c is a $(2k - 2)$ -coloring of G_e , then A is row colorful iff B is column colorful.*

Now we are prepared to construct G_e . Replace each x_j in the case $k = 2$ by a copy A_j of M and join each A_j to A_{j+1} by a connection set. Similarly

replace each y_j with B_j joined to B_{j+1} . If φ_e is a $(2k-2)$ -coloring which eventually colors both A_1 and B_1 so that they have the same parity, then φ_e must color A_i and B_i so that they have the same parity. We then complete the construction of G_e by adding copies A and B of M which play the role of x and y in the case $k=2$. At this point φ_e cannot be a $(2k-2)$ -coloring of G_e . The other possibilities are handled analogously. Note that whenever we introduce a new vertex in this construction we can specify both its adjacencies to previous vertices and its eventual degree. Thus G will be highly recursive. $\square$

A graph G is said to be *perfect* if $\chi(H) = \omega(H)$ for every induced subgraph H of G . Examples of perfect graphs include comparability graphs and co-comparability graphs of partial orders. Kierstead showed that in the case of perfect graphs Schmerl's technique could be considerably improved by searching farther in the graph before committing to the color of a vertex.

Theorem 2.3 (Kierstead [16]) *Every perfect, k -colorable, highly recursive graph is recursively $(k+1)$ -colorable. Moreover there exist perfect, k -colorable, highly recursive graphs which are not recursively k -colorable.*

Brooks' Theorem [2] states that, for any graph G , $\chi(G) \leq \Delta(G)$, unless $\Delta(G) < \omega(G)$, or both $\Delta(G) = 2$ and G contain an odd cycle. Schmerl proved that Brooks' Theorem is effective in the following sense.

Theorem 2.4 (Schmerl [36]) *Let G be a highly recursive graph. Then $\chi_{\text{rec}}(G) \leq \Delta(G)$, unless $\Delta(G) < \omega(G)$, or both $\Delta(G) = 2$ and G contain an odd cycle.*

Two edges d and e of a graph are adjacent if they share a vertex. In this case we write $d \sim e$. A k -edge coloring c of a graph G is a function $c : E \rightarrow C$ such that $|C| = k$ and if $d \sim e$ then $c(d) \neq c(e)$. In this case we say that G is k -edge colorable. The *chromatic index* of a graph G , denote $\chi'(G)$, is the least integer k such that G is k -edge colorable. A k -edge coloring f of G is *recursive* if f is a recursive function. In this case G is *recursively k -edge colorable*. The *recursive chromatic index* of G , denoted $\chi'_{\text{rec}}(G)$ is the least k such that G is recursively k -edge colorable. Vizing's Theorem states that, for any graph G , $\chi'(G) \leq \Delta(G) + 1$. Kierstead proved that Vizing's Theorem is almost effective.

Theorem 2.5 (Kierstead [16]) *Let G be a highly recursive graph. Then $\chi'_{\text{rec}}(G) \leq \chi'(G) + 1 (\leq \Delta(G) + 2)$.*

As stated the theorem is tight, since previously Manaster and Rosenstein had proved:

Theorem 2.6 (Manaster and Rosenstein [31]) *There exists a highly recursive graph G such that $\chi'(G) = \Delta(G)$, but $\chi'(G) < \chi'_{\text{rec}}(G)$.*

However, Vizing's Theorem may actually be effective. The most interesting open question concerning recursive coloring of highly recursive graphs is probably:

Question 2.7 *Is $\chi'_{\text{rec}}(G) \leq \Delta(G) + 1$ for every highly recursive graph G ?*

3 Recursive ordered sets

The first indication that there might be interesting statements to prove regarding recursive coloring of recursive graphs in general arose from the study of recursive ordered sets. An ordered set is a pair $P = (X, \leq)$, where $\leq$ is a reflexive, antisymmetric, transitive relation on X . The ordered set P is *recursive* if both X is a recursive set and $\leq$ is a recursive relation. If x and y are points in X such that either $x \leq y$ or $y \leq x$, then x and y are said to be *comparable*; otherwise, they are *incomparable*. The *comparability graph* of P is the graph $C(P) = (X, E)$ where two vertices x and y are adjacent iff they are comparable points in P . Similarly, the *co-comparability graph* of P is the graph $C^c(P) = (X, E^c)$ where two vertices x and y are adjacent iff they are incomparable points in P . A *chain* in P is a clique in $C(P)$ and an *antichain* in P is a clique in $C^c(P)$. The *height* of P is $h(P) = \omega(C(P))$ and the *width* of P is $w(P) = \omega(C^c(P))$. It is well known that $\chi(C(P)) = \omega(C(P))$, i.e., P can be partitioned into $h(P)$ antichains. Dilworth's Theorem may be formulated as $\chi(C^c(P)) = \omega(C^c(P))$, i.e., P can be partitioned into $w(P)$ chains. A *comparability (co-comparability) graph* is the comparability (co-comparability) graph of some ordered set. Because every induced subgraph of a comparability (co-comparability) graph is a comparability (co-comparability) graph, comparability (co-comparability) graphs are perfect. Kierstead answered a question of Schmerl with the following theorem:

Theorem 3.1 (Kierstead [15]) *Every recursive ordered set of width w can be partitioned into $(5^w - 1)/4$ recursive chains.*

Kierstead also constructed examples of recursive ordered sets of width w which could not be partitioned into $w^{5/3}$ recursive chains. Szemerédi and Trotter [40] improved the exponent to 2. Theorem 3.1 led Schmerl to ask the following question, which we shall answer in Section 5.

Question 3.2 (Schmerl [36]) *Does there exist a function f such that, for any recursive co-comparability graph G , $\chi_{\text{rec}}(G) \leq f(\omega(G))$?*

The reason Theorem 3.1 does not answer Schmerl's question is that one can construct recursive co-comparability graphs which are not the co-comparability graph of any recursive ordered set (see Penrice [33]). Moreover the algorithm in the proof of Theorem 3.1 makes explicit use of the order relation of the partial order, not just its comparability graph. In fact, Schmerl showed that for the complementary problem the order relation is crucial. Since every acyclic graph is the comparability graph of an ordered set of height two, Theorem 2.1 shows that there exist recursive comparability graphs G such that $\omega(G) = 2$ and $\chi_{\text{rec}}(G) = \infty$. In contrast, we have:

Theorem 3.3 (Schmerl [34]) *Every recursive ordered set of height h can be covered with $(h+1)h/2$ recursive antichains. Moreover, for all positive integers h , there exists a recursive ordered set of height h which cannot be covered with $(h+1)h/2 - 1$ recursive antichains.*

Proof. Let $P = (X, \leq)$ be a recursive ordered set of height h . Then the points of X are natural numbers. Say $X = \{x_1 \ll x_2 \ll \dots\}$, where $\ll$ is the standard order on the natural numbers. Let $X_i = \{x_1, \dots, x_i\}$,

$$U(x_i) = \{y \in X_i : x_i \leq y\} \quad \text{and} \quad D(x_i) = \{y \in X_i : x_i \geq y\}.$$

Let $u(x_i)$ and $d(x_i)$ be the length of the longest chain in $U(x_i)$ and $D(x_i)$, respectively. Notice that $u(x_i) + d(x_i) \leq h + 1$. Partition P into the sets A_{ud} , for $1 \leq u, d \leq h$, by assigning x_i to $A_{u(x_i)d(x_i)}$. Clearly the A_{ud} are recursive sets and there are at most $(h+1)h/2$ of them. It remains to show that they are antichains. Suppose x_j is in A_{ud} and $i < j$. If $x_i \leq x_j$, then $d(x_i) < d(x_j) = d$ and thus $x_i \notin A_{ud}$. Similarly, if $x_j \leq x_i$, then $u(x_i) < u(x_j) = u$ and thus $x_i \notin A_{ud}$. Thus A_{ud} is an antichain.

Next we construct a recursive ordered set $P = (X, \leq)$ of height h , which cannot be partitioned into $(h+1)h/2 - 1$ recursive antichains. We use a standard diagonal argument to construct P . The universe X will consist of disjoint parts $X_1, X_2, \dots$, one part for each of the countably many possibilities $\varphi_1, \varphi_2, \dots$ for partitioning P into recursive antichains. There will be no comparabilities between the parts X_e . Thus to insure that P has height at most h , it suffices to ensure that each part has height at most h . The e -th part will be constructed using a "wait and see" argument so that φ_e does not partition X_e into fewer than $(h+1)h/2$ antichains. Note that, in contrast to the case of highly recursive graphs, we do not have to add to X_e until φ_e assigns all the points currently in X_e to antichains. So it remains to show that we can construct a height h ordered set P_e on X_e by presenting the points of X_e one at a time, and at the time the i -th point is presented, determining its comparability relations to the previous points based only on information about the antichains to which φ_e has assigned the previous points.

The ordered set P_e will consist of chains $C_{11}, \dots, C_{1h}, C_{21}, \dots, C_{2h-1}, \dots, C_{h1}$, constructed in that order, such that:

- (0) $h(C_{ij}) \leq h+1-i$;
- (1) if $x \in C_{ij}$ and $y \in C_{st}$, with $(i, j) \neq (s, t)$, then x is incomparable to y , except that when $s < i$ and y is the maximum element of C_{st} , then $x < y$; and
- (2) for $i \in \{1, \dots, h\}$, if φ_e partitions P_e into antichains, then φ_e is forced to assign the maximum elements $m_{i1}, \dots, m_{ih+1-i}$ of $C_{i1}, \dots, C_{ih+1-i}$ to distinct antichains.

Note that by (0) and (1) P_e will have height h and by (1) and (2) φ_e will be forced to assign the $(h+1)h/2$ elements m_{ij} to distinct antichains. Each of the chains C_{ij} is constructed by continuing to add points, which satisfy (1), to the top of the existing chain, until (2) is satisfied. Note that we will need at most $j \leq h+1-i$ elements for C_{ij} . $\square$

An interval order is an ordered set whose points can be represented by intervals of the real line so that a point x is less than a point y iff the right end point of the interval which represents x is less than the left end point of the interval which represents y . An interval graph is the co-comparability graph of an interval order. Kierstead and Trotter answered Schmerl's question for the special case of interval orders.

Theorem 3.4 (Kierstead and Trotter [28]) *For every recursive interval graph G , $\chi_{\text{rec}}(G) \leq 3\omega(G) - 2$. Moreover, for every ω there exists a recursive interval graph G with $\omega(G) = \omega$ such that $\chi_{\text{rec}}(G) = 3\omega - 2$.*

The following question remains open and is of great interest.

Questions 3.5 (Kierstead [15]) *Does there exist a polynomial p such that every recursive ordered set of width w can be covered by $p(w)$ recursive chains?*

Finally we mention one application of Theorem 3.1 to the dimension theory of ordered sets. A *linear order* is an ordered set such that all points are pairwise comparable. A *d -realizer* of an ordered set $P = (X, \leq)$ is a collection of d linear orders $\Sigma = \{L_1 = (X, \leq_1), \dots, L_d = (X, \leq_d)\}$ such that $x \leq y$ iff $x \leq_i y$ for $i \in \{1, \dots, d\}$. The realizer Σ is recursive if each of the L_i are recursive. The (*recursive*) *dimension* of an ordered set P is the least integer d such that there exists a (recursive) d -realizer of P . A *crown* is a height two ordered set on at least six points whose comparability graph is a cycle. It is not too difficult to prove that the dimension of an ordered set of width w is at most w (see [14]). The recursive dimension of a recursive ordered set of width w can be infinite; however, Kierstead, McNulty, and Trotter proved:

Theorem 3.6 (Kierstead, McNulty, and Trotter [22]) *Let P be a recursive ordered set of width w , which does not contain a crown. Then the recursive dimension of P is at most $((5^w - 1)/4)!$.*

4 On-line coloring

An *on-line graph* is a structure $G^{\ll}$ where $G = (V, E)$ is a graph and $\ll$ is a linear order on V . The linear order $\ll$ is called a *presentation* of G . Let $G_n^{\ll}$ denote the on-line graph induced by the $\ll$ -first n elements $V_n = \{v_1 \ll \dots \ll v_n\}$ of V . An algorithm (Turing machine) **A** for coloring the vertices of an on-line graph is said to be an *on-line algorithm* if the color of the n -th vertex v_n is determined solely by the isomorphism type of $G_n^{\ll}$. Intuitively, the algorithm **A** colors the vertices of $G^{\ll}$ one vertex at a time in the externally determined order $v_1 \ll v_2 \ll \dots$, and at the time a color is irrevocably assigned to v_n , the algorithm can only see $G_n^{\ll}$. The number

of colors that $\mathbf{A}$ uses to color $G^{\ll}$ is denoted by $\chi_{\mathbf{A}}(G^{\ll})$. The maximum of $\chi_{\mathbf{A}}(G^{\ll})$ over all on-line presentations $\ll$ is denoted by $\chi_{\mathbf{A}}(G)$. If Γ is a class of graphs, the maximum of $\chi_{\mathbf{A}}(G)$ over all G in Γ is denoted by $\chi_{\mathbf{A}}(\Gamma)$. The *on-line chromatic number* of Γ , denoted by $\chi_{\text{ol}}(\Gamma)$, is the minimum of $\chi_{\mathbf{A}}(\Gamma)$ over all on-line algorithms $\mathbf{A}$. A simple but important example of an on-line algorithm is the algorithm First-Fit, which colors the vertices of G with an initial sequence of the colors $\{1, 2, \dots\}$ by assigning to the vertex v the least color that has not already been assigned to any vertex adjacent to v .

Let G be a recursive graph. Then the vertices of G are integers. Let $\ll$ be the presentation of G in which the vertices have their natural order. Then the coloring of $G^{\ll}$ produced by an on-line algorithm $\mathbf{A}$ is a recursive coloring and so $\chi_{\text{rec}}(\Gamma) \leq \chi_{\text{ol}}(\Gamma)$. There are several technical problems which prevent us from reversing this inequality in general. However, all our lower bounds for on-line chromatic number will have easy translations to lower bounds for recursive chromatic number. Perhaps it would be interesting to develop a general formulation of this equivalence.

When the on-line chromatic number of a class of graphs Γ is finite it is the same as the on-line chromatic number of the class $F(\Gamma)$ consisting of all finite induced subgraphs of graphs in Γ . In the remainder of this article we restrict our attention to on-line algorithms for coloring finite on-line graphs. This makes for a more interesting theory since we can discuss bounds on the on-line chromatic number of a graph in terms of the cardinality of its vertex set.

Let Γ_k be the class of finite k -colorable graphs and

$$\Gamma_k(n) = \{(V, E) \in \Gamma_k : |V| \leq n\}.$$

We first consider bipartite (2-colorable) graphs. Here the situation is well understood. We first prove a more precise version of Theorem 2.1.

Theorem 4.1 *For every on-line algorithm $\mathbf{A}$ and positive integer t , there exists an on-line tree $T^{\ll}$ with 2^{t-1} vertices such that $\chi_{\mathbf{A}}(T^{\ll}) \geq t$.*

Proof. Argue by induction on t . The base step $t = 1$ is trivial, so consider the case $t = s + 1$. We claim that, by the induction hypothesis, there exists an on-line acyclic graph $F^{\ll}$ with components $T_1^{\ll}, \dots, T_s^{\ll}$ such that $\mathbf{A}$ uses i colors on T_i and T_i has 2^{i-1} vertices. The existence of $T_1^{\ll}$ follows from

the induction hypothesis applied to $\mathbf{A} = \mathbf{A}_1$. The existence of $T_1^{\ll} \cup T_2^{\ll}$ follows from the induction hypothesis applied to $\mathbf{A}_2$, the on-line algorithm which colors an on-line graph $G^{\ll}$ in the same way that $\mathbf{A}$ would color $G^{\ll}$ after first coloring $T_1^{\ll}$, etc.. Choose a set of vertices $Y = \{y_1, \dots, y_s\}$ such that $y_i \in V(T_i)$ and $\mathbf{A}$ assigns different colors to each vertex in Y . Now we obtain T_i by adding one last vertex x to $F^{\ll}$ so that, for $i \in \{1, \dots, s\}$, x is adjacent to y_i . Then $\mathbf{A}$ uses t colors on the set $Y \cup \{x\}$. $\square$

Lovasz, Saks, and Trotter found a nearly optimal on-line algorithm for the bipartite case. This combined with Theorem 4.1 yields

$$-1 + \log n \leq \chi_{\mathbf{A}}(\Gamma_2(n)) \leq 1 + 2 \log n.$$

Theorem 4.2 (Lovasz, Saks, and Trotter [30]) *There exists an on-line algorithm $\mathbf{A}$ such that for every 2-colorable graph G on n vertices, $\chi_{\mathbf{A}}(G) \leq 1 + 2 \log n$.*

Proof. Let $G^{\ll} = (V, E)$ be an on-line graph, where v_i is the i -th vertex in the presentation $\ll$. When v_i is presented there is a unique partition (I_1, I_2) of the connected component of v_i in $G_i^{\ll}$ into independent sets such that v_i is in I_1 . The algorithm $\mathbf{A}$ assigns v_i the least color not already assigned to some vertex of I_2 .

It suffices to show that if $\mathbf{A}$ uses at least t colors on any connected component of $G_i^{\ll}$, then that connected component contains at least $2^{\lfloor t/2 \rfloor}$ vertices. We argue by induction on i and note that the base step is trivial. For the induction step, observe that if $\mathbf{A}$ assigns v_i color $k+2$, then $\mathbf{A}$ must already have assigned color k to some vertex $v_p \in I_2$ and color $k+1$ to some other vertex in I_2 . Thus $\mathbf{A}$ must have assigned color k to some vertex $v_q \in I_1$. Since $\mathbf{A}$ assigned v_p and v_q the same color, v_p and v_q must be in separate components of $G_r^{\ll}$, where $r = \max\{p, q\}$. Thus, by the induction hypothesis, each of these connected components must have at least $2^{\lfloor k/2 \rfloor}$ vertices and so the component of v_i in $G_i^{\ll}$ has at least $2^{\lfloor (k+2)/2 \rfloor}$ vertices. $\square$

The situation in general is much harder. Already in the case of 3-colorable graphs very little is known despite several clever ideas and hard arguments. The following example of Szegedy shows that when n is relatively small compared to k , then no on-line algorithm can do very well on all graphs in $\Gamma_k(n)$.

Theorem 4.3 (Szegedy [39]) *For every on-line algorithm $\mathbf{A}$ and positive integer k , there exists an on-line graph $G^{\ll} \in \Gamma_k(k2^k)$ such that $\chi_{\mathbf{A}}(G^{\ll}) \geq 2^k - 1$.*

Proof. We construct $G^{\ll}$ in stages; $G_s^{\ll}$ is constructed at the s -th stage, which consists of three steps. First we introduce a new vertex v_s together with all edges from v_s to previous vertices. Next we determine the color $\mathbf{A}(v_s) = c$, which $\mathbf{A}$ assigns v_s . We may assume that $c \in \{c_1, \dots, c_{2^k-1}\}$. Finally we assign a color $f(v_s) \in \{r_1, \dots, r_k\}$ to v_s . Let C_j be the set of vertices that $\mathbf{A}$ has colored c_j , let R_i be the set of vertices that we have colored r_i , and let $X_{ij} = R_i \cap C_j$. We shall try to maintain the following induction hypothesis:

- (1) f is a k -coloring, and
- (2) $|X_{ij}| \leq 1$ for all $i \in [k]$ and $j \in [2^k - 1]$.

By (1), $G^{\ll}$ is k -colorable; and by (2), $G^{\ll}$ has less than $k2^k$ vertices. Thus it suffices to show that we can maintain (1) and (2) until $\mathbf{A}$ uses $2^k - 1$ colors.

Let $S \subset \{1, \dots, k\} = [k]$. We say that S is *represented* if there exists j such that $i \in S$ iff $X_{ij} \neq \emptyset$. If every non-empty subset of $[k]$ is represented then $\mathbf{A}$ has already used $2^k - 1$ colors and we are done. Otherwise, suppose S is a nonempty subset of $[k]$ which is not represented. Let v_s be adjacent to v iff $v \in R_i$ and $i \notin S$. Suppose $\mathbf{A}$ colors v_s with c_j . Then $X_{ij} = \emptyset$ for all $i \notin S$. Thus, since S is neither empty nor represented, there exists $i \in S$ such that $X_{ij} = \emptyset$. Let $f(v_s) = i$. $\square$

Note that for $n \leq k2^k$, the above argument shows that for every on-line coloring algorithm $\mathbf{A}$, there exists a k -colorable on-line graph $G^{\ll}$ on n vertices such that $\mathbf{A}$ uses at least n/k colors on one color class of some k -coloring of $G^{\ll}$. With a little extra care, Vishwanathan used the idea of the proof of Theorem 4.1 to prove:

Theorem 4.4 (Vishwanathan [41]) *For every on-line algorithm $\mathbf{A}$, for all positive integers k , and for arbitrarily large n , there exists an on-line graph $G^{\ll} \in \Gamma_k(n)$ such that*

$$\frac{((-1 + \log n)/(2k + 1))^{k-1}}{(k - 1)} \leq \chi_{\mathbf{A}}(G^{\ll}).$$

Proof. For fixed parameters $c, t \geq 1$, let

$$n(a, b) = 2^{2cat} (2^{c(b+1)} - 2^c),$$

$$\chi(a, b) = \frac{b}{(a-1)} t^{a-2} \quad \text{for } a > 1, \text{ and}$$

$$\chi(1, b) = 1.$$

(For this proof c will be 1, but later we will change the value of c in the proof of Theorem 4.13.) We shall recursively construct classes $\Gamma(a, b)$ consisting of pairs $(G^{\ll}, I)$, where $G^{\ll}$ is an on-line graph and I is a distinguished independent set. The sets $\Gamma(a, b)$ will have the following properties:

- (1) for all $(G^{\ll}, I)$ in $\Gamma(a, b)$, $G^{\ll}$ has at most $n(a, b)$ vertices;
- (2) for all $(G^{\ll}, I)$ in $\Gamma(a, b)$, $G^{\ll}$ can be a -colored so that I is contained in a color class; and
- (3) for every on-line algorithm $\mathbf{A}$, there exists a pair $(G^{\ll}, I)$ in $\Gamma(a, b)$ such that $\mathbf{A}$ uses at least $\chi(a, b)$ colors on the vertices of I .

For all b let $\Gamma(1, b) = \{(G^{\ll}, I)\}$, where $I = V(G)$ consists of a single vertex. Then (1), (2) and (3) are satisfied. Now suppose that, for every on-line algorithm $\mathbf{A}$ and integer b we can construct $\Gamma(a-1, b)$ satisfying (1), (2) and (3). Let $\Gamma(a, 1) = \Gamma(a-1, t)$. Then (2) holds. Clearly (1) and (3) are satisfied by $(G^{\ll}, I) \in \Gamma(2, 1)$. Suppose $a > 2$. Then (1) and (3) hold, since

- (i) for all $(G^{\ll}, I) \in \Gamma(a, 1)$,

$$\begin{aligned} |V(G)| &\leq n(a-1, t) = 2^{2c(a-1)t} (2^{c(t+1)} - 2^c) \\ &\leq 2^{2cat} (2^{2c} - 2^c) = n(a, 1), \end{aligned}$$

and

- (ii) $\mathbf{A}$ uses at least $\chi(a-1, t)$ colors on I while coloring $G^{\ll}$, for some $(G^{\ll}, I) \in \Gamma(a, 1)$, and

$$\chi(a-1, t) = \frac{t}{(a-2)} t^{a-3} \geq \frac{1}{(a-1)} t^{a-2} = \chi(a, 1).$$

Finally, suppose that we have constructed $\Gamma(a, b-1)^\ll$. Let

$$G^\ll + H^\ll + B^\ll + E$$

denote the on-line graph formed from disjoint copies of the on-line graphs $G^\ll, H^\ll$ and $B^\ll$ by adding the edges in E to the union of the copies, and presenting all the vertices in G, H and B according to $\ll$, with the vertices of G followed by the vertices of H , followed by the vertices of B . Let $\Gamma(a, b)$ consist of all pairs of the form $(G^\ll + H^\ll + B^\ll + E, J)$, where

$$(G^\ll, I_1), (H^\ll, I_2) \in \Gamma(a, b-1),$$

$$(B^\ll, I_3) \in \Gamma(a-1, 2t),$$

$$E = \{iv : i \in I_1 \text{ and } v \in V(B)\}, \quad \text{and}$$

$$J \in \{I_1 \cup I_2, I_2 \cup I_3\}.$$

First note that for $J \in \{I_1 \cup I_2, I_2 \cup I_3\}$, $(G^\ll + H^\ll + B^\ll + E, J)$ can be a -colored so that J is contained in a color class. Thus (2) holds. Arguing as in the proof of Theorem 4.1, and using the induction hypothesis, for any on-line algorithm **A** there exist pairs

$$(G^\ll, I_1), (H^\ll, I_2) \in \Gamma(a, b-1) \quad \text{and} \quad (B^\ll, I_3) \in \Gamma(a-1, 2t)$$

such that **A** uses at least $\chi(a, b-1)$ colors on each of I_1 and I_2 and **A** uses $\chi(a-1, 2t)$ colors on I_3 . If **A** does not use $\chi(a, b-1) + \frac{1}{2}\chi(a-1, 2t)$ colors on $I_1 \cup I_2$, then **A** uses the same $\chi(a, b) - \frac{1}{2}\chi(a-1, 2t)$ colors on both I_1 and I_2 . Since every vertex of I_3 is adjacent to every vertex of I_1 , none of these colors are used on I_3 . Thus **A** uses $\chi(a, b-1) + \frac{1}{2}\chi(a-1, 2t)$ colors on $I_2 \cup I_3$. Thus (3) holds, since

$$\begin{aligned} \text{(iii)} \quad \chi(a, b-1) + \frac{1}{2}\chi(a-1, 2t) &\geq \frac{(b-1)}{(a-1)}t^{a-2} + \frac{1}{2}\frac{2t}{(a-2)}t^{a-3} \\ &\geq \frac{b}{(a-1)}t^{a-2} = \chi(a, b). \end{aligned}$$

Finally, (1) holds since

$$\begin{aligned}
 \text{(iv)} \quad |V(G)| &\leq 2n(a, b-1) + n(a-1, 2t) \\
 &\leq 2 * 2^{2cat} (2^{cb} - 2^c) + 2^{2c(a-1)t} (2^{c(2t+1)} - 2^c) \\
 &\leq 2^{2cat} [2(2^{cb} - 2^c) + 2^c] \\
 &\leq 2^{2cat} (2^{c(b+1)} - 2^c) = n(a, b).
 \end{aligned}$$

To prove the theorem, fix a size s . Choose $b = t$ so that $n = n(k, b) \geq s$. Then for each $(G^{\ll}, I)$ in $\Gamma(k, t)$, $G^{\ll}$ has at most n vertices. For some such $G^{\ll}$,

$$\chi_{\mathbf{A}}(G^{\ll}) \geq \frac{t^{k-1}}{(k-1)}.$$

Also $\log n \leq c(2kt + t + 1)$. Thus

$$\chi_{\mathbf{A}}(G^{\ll}) \geq \frac{((-c + \log n)/(2ck + c))^{k-1}}{(k-1)}.$$

Taking $c = 1$ we are done. $\square$

The first sublinear upper bound on $\chi_{\text{ol}}(\Gamma_k(n))$ was obtained by Lovasz, Saks, and Trotter. When k is not fixed, Szegedy's example shows that their upper bound is reasonably tight. However when k is fixed there is a huge gap. Here $\log^{(k)}$ denotes log iterated k times and $\log^* n$ is the least integer t such that $\log^{(t)} n \leq 1$.

Theorem 4.5 (Lovasz, Saks, and Trotter [30]) *For every integer $k \geq 3$ there exists an on-line algorithm $\mathbf{A}$ such that for every on-line k -colorable graph $G^{\ll}$ on n vertices, $\chi_{\mathbf{A}}(G^{\ll}) = O(n \log^{(2k-3)} n / \log^{(2k-4)} n)$. Moreover, there exists an on-line algorithm $\mathbf{B}$ such that for every on-line graph $G^{\ll}$ on n vertices $\chi_{\mathbf{B}}(G^{\ll}) / \chi(G) \leq O((n / \log^* n))$.*

In some sense their algorithm is a complicated on-line implementation of the following (off-line) polynomial time algorithm $\mathbf{A}(k, t)$ for coloring 3-colorable graphs, which we call Modified First-Fit. First color $G = (V, E)$ with First-Fit. Save the first t color classes $C_1, \dots, C_t$. Let

$$X = V - \bigcup_{1 \leq i \leq t} C_i.$$

Each vertex in X is adjacent to a vertex in each of the C_i . Fix i and let $C_i = \{v_{i1}, \dots, v_{is(i)}\}$. Partition X into *residue* sets $R_{i1}, \dots, R_{is(i)}$, where

$$R_{ij} = \{x \in X : v_{ij} \sim x \text{ and not } v_{ih} \sim x \text{ for any } h < j\}.$$

Using disjoint color sets, 2-color each R_{ij} in polynomial time. This is possible since $R_{ij} \subset N(v_{ij})$. Thus we obtain a $(t + 2s(i))$ -coloring of G . Letting $t = n^{1/2}$ and choosing i so that $|C_i|$ is as small as possible, we obtain a $(3n^{1/2})$ -coloring. The problem in implementing this algorithm on-line is that we do not know ahead of time which color class C_i will be minimum. Thus we must hedge our bets.

We shall need the following easy combinatorial lemma.

Lemma 4.6 *Let δ be a constant with $0 < \delta < 1/2$. Let F be a family of subsets of a finite set S such that both*

- (i) $|R| \geq \delta|S|$, and
- (ii) $|R \cap Q| < \delta^2|S|/2$, for all $R, Q \in F$.

Then $|F| \leq 2/\delta$.

Proof of Theorem 4.5. The second statement is an easy consequence of the first. We shall sketch the proof of the first.

Let $G^{\ll} = (V, E)^{\ll}$ be an on-line graph. For fixed positive integers d and m with $d < m/2$ we make the following definitions. Let $\varepsilon_1 = d/m$ and $\varepsilon_{s+1} = \varepsilon_s^2/2$. Then $\varepsilon_s = 2(d/(2m))^{2^{s-1}}$. An r -subset $R \subset V_m$ is *legal* if either $R = \emptyset$ or $|\cap_{v_s \in R} N(v_s) \cap V_s| \geq \varepsilon_r m$. Using these parameters, we define an on-line procedure **Partition**(d, m, f), where f is a d -coloring of $G_i^{\ll}$ for some $i < m$ with color classes $\{C_1, \dots, C_d\}$. **Partition**(d, m, f) partitions the vertices of $G_m^{\ll}$ into independent sets $\{D_1, \dots, D_d\}$, with $C_i \subset D_i$, and legal residue sets $\{\emptyset, R_1, R_2, \dots, R_x\}$, for some natural number x . When a new vertex v_s is presented **Partition**(d, m, f) assigns v_s to:

- (1) D_i , for some i such that $D_i \cup \{v_s\}$ is independent, if possible; otherwise
- (2) R_i , where R_i is maximum subject to $R_i \cup \{v_s\}$ being legal.

Notice that if (1) fails for v_s then $\{v_s\}$ is legal and so it is possible to satisfy (2). Also, each residue class is contained in the neighborhood of a single vertex and thus is $(k-1)$ -colorable.

Claim 1 *When **Partition**(d, m, f) terminates there are at most $2/\varepsilon_t$ residue sets of size t .*

Proof. We shall apply Lemma 4.6 with $\delta = \varepsilon_t$. Consider the family $F = \{W_i : W_i = \cap_{v_s \in R_i} N(v_s) \cap V_s \text{ and } |R_i| = t\}$. Since R_i is legal, (i) of Lemma 4.6 holds. To see that (ii) of Lemma 4.6 holds, suppose that W_i and W_j are in F , v was the last vertex added to $R_i \cup R_j$, and v was added to R_j . Then $R_i \cup \{v\}$ was not legal, and thus $|W_i \cap W_j| < \varepsilon_{t+1}m = \delta^2 m/2$. Thus $|F| \leq 2/\varepsilon_t$. $\square$

Claim 2 *When **Partition**(d, m, f) terminates there are at most $1/\varepsilon_t + m/t$ residue sets for any $t \geq 1$.*

Proof. Since the residue sets are pairwise disjoint, there are at most m/t residue sets of size at least t . Thus, using Claim 1, the number of residue sets is bounded by $m/t + \sum_{1 \leq s < t} 2/\varepsilon_s \leq m/t + 1/\varepsilon_t$. $\square$

Claim 3 *If $d \geq m/(\log \log m)$, where $m \geq 4$, then **Partition**(d, m, f) creates at most $3m/(\log \log m)$ residue sets.*

Proof. Let $t = \frac{1}{2} \log \log m$. Then, using Claim 2, we are done, since

$$\begin{aligned} m/t + 1/\varepsilon_t &= 2m/\log \log m + 2(d/(2m))^{-2^{t-1}} \\ &\leq 2m/\log \log m + 2(2 \log \log m)^{(\log m)^{1/2}} \\ &\leq 3m/\log \log m. \end{aligned} \quad \square$$

Now we are nearly done. First suppose $k = 3$ and we know n ahead of time. Let f be the First-Fit coloring of an initial sequence of vertices of $G^{\ll}$, which uses $d = n/(\log \log n)$ colors. We extend f by using the on-line procedure **Color**(d, n, f) defined as follows. First apply **Partition**(d, n, f). Every time an element is added to a residue set, color it using the on-line algorithm from Theorem 4.2. By Claim 3, at most $x = 3n/(\log \log n)$ residue sets are created. The total number of colors used on these residue sets is at most

$$\sum_{1 \leq i \leq x} (1 + 2 \log |R_i|) \leq 3x \log(n/x) \leq 9n \log^{(3)} n / \log^{(2)} n.$$

Thus the total number of colors used is at most $10n \log^{(3)} n / \log^{(2)} n$. In general, arguing by induction on k , a similar calculation holds. Finally we must

deal with not knowing the number of vertices n in advance. Begin by guessing $n = 100$ and run the algorithm above for $n = 100$ on the first 100 vertices. If n turns out to be greater than 100, let $d_1 = 10n \log^{(3)} n / \log^{(2)} n$, f the coloring created so far, and n_1 the greatest integer such that $n_1 / (\log \log n_1) \leq d_1$. Then apply **Color**(d_1, n_1, f). If n turns out to be bigger than n_1 , then choose d_2 and n_2 in a similar manner and apply **Color**(d_2, n_2, f), etc.. It is easy to see that this process will require at most $100n \log^{(3)} n / \log^{(2)} n$ colors. $\square$

Very recently the author has obtained the following results. Lemma 4.6 still plays a crucial role in their proofs.

Theorem 4.7 (Kierstead [21]) *For every positive integer k , there exists an on-line algorithm A_k and an integer N such that, for every on-line k -colorable graph $G^{\ll}$ on $n \geq N$ vertices, $\chi_{A_k}(G^{\ll}) \leq n^{1-1/k!}$.*

For the special cases $k = 3$ and $k = 4$ we obtain the following stronger results.

Theorem 4.8 (Kierstead [21]) *There exists an on-line algorithm A_3 such that, for every on-line 3-colorable graph $G^{\ll}$ on n vertices, $\chi_{A_3}(G^{\ll}) < 20n^{2/3}(\log n)^{1/3}$.*

Theorem 4.9 (Kierstead [21]) *There exists an on-line algorithm A_4 such that, for every on-line 4-colorable graph $G^{\ll}$ on n vertices, $\chi_{A_4}(G^{\ll}) < 120n^{5/6}(\log n)^{1/6}$.*

Problem 4.10 *Determine (or at least find better bounds for) the on-line chromatic number of the class of 3-colorable graphs.*

Vishwanathan recognized that Modified First-Fit was an algorithm ripe for randomization. A *randomized* on-line algorithm is a probability space Ω whose points are on-line algorithms. For an on-line graph $G^{\ll}$, let $\chi_{\Omega}(G^{\ll})$ denote the expected value of the random variable $\chi_{\mathbf{A}}(G^{\ll})$. Vishwanathan proved the following very nice theorems.

Theorem 4.11 (Vishwanathan [41]) *There exists a randomized on-line algorithm Ω such that for every on-line graph $G^{\ll}$ on n vertices, $\chi_{\Omega}(G^{\ll}) \leq O(\chi \cdot 2^{\chi} n^{(\chi-2)/(\chi-1)} (\log n)^{1/(\chi-1)})$, where $\chi = \chi(G)$.*

Proof. We shall only prove the case $\chi = 3$. The general argument is similar. As in the proof of Theorem 4.5, we can assume we know n ahead of time. We use a randomized on-line version of Modified First-Fit. The points of Ω will be the on-line algorithms $\mathbf{A}'(1, t), \dots, \mathbf{A}'(t, t)$, where $t = (n \log n)^{1/2}$, each having the same probability. The primes indicate that the residue sets are colored with the on-line algorithm from Theorem 4.2. Then, for any on-line graph $G^{\ll}$,

$$\begin{aligned}
 \chi_{\Omega}(G^{\ll}) &= \frac{1}{t} \sum_{1 \leq i \leq t} \chi_{\mathbf{A}(i, t)}(G^{\ll}) \\
 &\leq \frac{1}{t} \sum_{1 \leq i \leq t} \left(t + \sum_{1 \leq j \leq s(i)} 3 \log |R_{ij}| \right) \\
 &\leq t + \frac{3}{t} \sum_{1 \leq i \leq t} \sum_{1 \leq j \leq s(i)} \log |R_{ij}| \\
 &\leq t + \frac{3 \log t}{tn} \\
 &\leq 4(n \log n)^{1/2}, \quad \text{since}
 \end{aligned}$$

$$\sum_{1 \leq i \leq t} \sum_{1 \leq j \leq s(i)} |R_{ij}| \leq tn, \text{ and } |\{(i, j) : 1 \leq i \leq t, 1 \leq j \leq s(i)\}| \leq n. \quad \square$$

Vishwanathan's result can be stated in terms of parallel computation as follows. There exists a collection of $(n \log n)^{1/2}$ on-line algorithms such that for any 3-colorable on-line graph $G^{\ll}$, one of the algorithms uses at most $4(n \log n)^{1/2}$ colors to color $G^{\ll}$.

Vishwanathan also obtained lower bounds on the performance of randomized on-line coloring algorithms. His proof uses the following weak version of a lemma of Yao.

Lemma 4.12 (Yao [43]) *Let Γ be a discrete probability space whose points are on-line graphs. If, for every on-line algorithm $\mathbf{A}$, the expected value $E_{\Gamma}(\chi_{\mathbf{A}}(G^{\ll}))$ of the random variable $\chi_{\mathbf{A}}(G^{\ll})$ is at least b , then for every randomized on-line algorithm Ω , there exists an on-line graph $G^{\ll}$ in Γ such that $\chi_{\Omega}(G^{\ll}) \geq b$.*

Proof. By the pigeon hole principle, it suffices to show that the expected value $E_\Gamma(\chi_\Omega(G))$ is at least b . To see this, note that

$$\begin{aligned} E_\Gamma(\chi_\Omega(G)) &= \sum_{G \ll \Gamma} \sum_{\mathbf{A} \in \Omega} \chi_{\mathbf{A}}(G^{\ll}) \Pr_\Gamma(G^{\ll}) \Pr_\Omega(\mathbf{A}) \\ &= \sum_{\mathbf{A} \in \Omega} E_\Gamma(\chi_{\mathbf{A}}(G^{\ll})) \Pr_\Omega(\mathbf{A}) \geq b. \quad \square \end{aligned}$$

Theorem 4.13 (Vishwanathan [41]) *For every randomized on-line coloring algorithm Ω , for all positive integers k , and for arbitrarily large n , there exists an on-line graph $G^{\ll}$ on n vertices with $\chi(G) = k$ such that $\chi_\Omega(G^{\ll})$ is at least*

$$\frac{((-4 + \log n)/(8k + 4))^{k-1}}{(2k - 2)}.$$

Proof. By Lemma 4.12 it suffices to construct a discrete probability space $\Gamma(n, k)$ whose points are k -colorable on-line graphs on at most n vertices such that $E_\Gamma(\chi_{\mathbf{A}}(G^{\ll}))$ is at least $((-4 + \log n)/(8k + 4))^{k-1}/(2k - 2)$, for every on-line algorithm $\mathbf{A}$. The construction is very similar to the construction in the proof of Theorem 4.4. For fixed parameters c and t , let $n(a, b) = 2^{2cat} (2^{c(b+1)} - 2^c)$, $\chi(a, b) = (b/(a - 1)) t^{a-2}$ for $a > 1$, and $\chi(1, b) = 1$. We shall recursively construct discrete probability spaces $\Gamma(a, b)$, whose points are pairs $(G^{\ll}, I)$, where I is an independent set in the on-line graph $G^{\ll}$.

Each pair will have the same probability. For an on-line algorithm $\mathbf{A}$, we will say that a pair $(G^{\ll}, I)$ in $\Gamma(a, b)$ is $\mathbf{A}$ -good if $\mathbf{A}$ uses at least $f(a, b)$ colors on I while coloring $G^{\ll}$. We require that $\Gamma(a, b)$ have the following properties:

- (i) for all $(G^{\ll}, I)$ in $\Gamma(a, b)$, $G^{\ll}$ has at most $n(a, b)$ vertices;
- (ii) for all $(G^{\ll}, I)$ in $\Gamma(a, b)$, $G^{\ll}$ can be a -colored so that I is contained in a color class; and
- (iii) for every on-line algorithm $\mathbf{A}$, the probability that $(G^{\ll}, I)$ is $\mathbf{A}$ -good is at least $1/2$.

The recursive construction of the $\Gamma(a, b)$ is exactly the same as in the proof of Theorem 4.4, except for the secondary induction step. Suppose we have

constructed $\Gamma(a-1, b')$ for all b' and $\Gamma(a, b-1)$. Let $\Gamma'(a, b)$ consist of all pairs of the form $(G^{\ll} + H^{\ll} + B^{\ll} + E, J)$ where $(G^{\ll}, I_1), (H^{\ll}, I_2) \in \Gamma(a, b-1)$, $(B^{\ll}, I_3) \in \Gamma(a-1, 2t)$, $E = \{ib : i \in I_1 \text{ and } b \in V(B)\}$, and $J \in \{I_1 \cup I_2, I_2 \cup I_3\}$. Let $\Gamma(a, b)$ consist of all pairs of the form $(G^{\ll}, I) = (G_1^{\ll} + \cdots + G_{16}^{\ll} + \emptyset, I_1 \cup \cdots \cup I_{16})$, where $(G_i^{\ll}, I_i)$ is in $\Gamma'(a, b)$.

Clearly (2) holds. Consider $(G^{\ll}, I) = (G^{\ll} + H^{\ll} + B^{\ll} + E, J)$ in $F'(a, b)$. If $(G^{\ll}, I_1), (H^{\ll}, I_2), (B^{\ll}, I_3)$ are all **A**-good, then the probability that $(G^{\ll}, I)$ is **A**-good is at least $1/2$. Since the former events are independent, and have probability at least $1/2$ by the induction hypothesis, the probability that $(G^{\ll}, I)$ is **A**-good is at least $1/16$. It follows easily that the probability that a pair in $\Gamma(a, b)$ is **A**-good is at least $1/2$ and so (3) holds.

Finally, (1) holds since, for all $(G^{\ll}, I)$ in $\Gamma(a, b)$,

$$\begin{aligned}
 \text{(iv)} \quad |V(G)| &\leq 16n(a, b-1) + n(a-1, 2t) \\
 &\leq 16 * 2^{2cat} (2^{cb} - 2^c) + 2^{2c(a-1)t} (2^{c(2t+1)} - 2^c) \\
 &\leq 2^{2cat} [16(2^{cb} - 2^c) + 2^c] \\
 &\leq 2^{2cat} (2^{c(b+1)} - 2^c) = n(a, b), \text{ provided that } c = 4.
 \end{aligned}$$

As before, an easy calculation now shows that $E_{\Gamma}(\chi_{\mathbf{A}}(G^{\ll}))$ is at least $((-4 + \log n)/(8k + 4))^{k-1}/(2k - 2)$, and we are done. $\square$

Finally we mention a recent theorem of Irani [12, 13]. A graph $G = (V, E)$ is d -inductive if the vertices of G can be ordered by $v_1 < v_2 < \cdots < v_n$ so that for every index i , $|\{j < i : v_i \sim v_j\}| \leq d$. For example, acyclic graphs are 1-inductive, and planar graphs are 5-inductive. A chordal graph G is $\omega(G)$ -inductive. Clearly the chromatic number of a d -inductive graph G is at most $d+1$, since First-Fit will use only $(d+1)$ colors to color $G^{\ll}$, where $<$ is the presentation that witnesses that G is d -inductive. Irani showed that First-Fit performs well on any presentation.

Theorem 4.14 (Irani [12, 13]) *If G is a d -inductive graph on n vertices, then First-Fit uses at most $O(d \log n)$ colors to color any on-line presentation $G^{\ll}$ of G . Moreover, for any on-line algorithm **A**, there exists a d -inductive on-line graph $G^{\ll}$ such that **A** uses at least $\Omega(d \log n)$ colors to color $G^{\ll}$.*

5 On-line χ -bounded classes

In the previous section we bounded the on-line chromatic number of graphs on n vertices with fixed chromatic number in terms of n . In this section we shall introduce classes of graphs whose on-line chromatic number can be bounded solely in terms of their clique number (and thus also in terms of their chromatic number). A class of graphs Γ is said to be χ -bounded if there exists a function f such that $\chi(G) \leq f(\omega(G))$ for every graph $G \in \Gamma$. In this case f is called a *binding* function. Note that the class of perfect graphs is an example of a χ -bounded class. Similarly, Γ is *on-line χ -bounded* if there exists a function f and an on-line algorithm $\mathbf{A}$ such that $\chi_{\mathbf{A}}(G) \leq f(\omega(G))$, for every $G \in \Gamma$. If First-Fit witnesses that a class is on-line χ -bounded the class is said to be First-Fit χ -bounded. For a graph H , let $\text{Forb}(H)$ denote the class of graphs which do not contain an induced subgraph isomorphic to H .

At the same time that logicians were studying the recursive chromatic number of highly recursive graphs two graph theorists, Gyárfás, and independently Sumner, proposed the following conjecture.

Conjecture 5.1 (Gyárfás [6] and Sumner [38]) *For every tree T , the class of graphs $\text{Forb}(T)$ is χ -bounded.*

The *girth* of a graph which contains a cycle is the number of vertices in the smallest cycle it contains. If H is a graph which contains a cycle, then $\text{Forb}(H)$ is not χ -bounded, since Erdős and Hajnal [5] have shown that there are graphs with arbitrarily large chromatic number and girth (in particular, girth larger than the girth of H). Such graphs are clearly in $\text{Forb}(H)$ and have clique number 2. If F is an acyclic graph with connected components (trees) $T_1, \dots, T_c$, then it is not hard to show that $\text{Forb}(F)$ is χ -bounded iff each of the trees $T_1, \dots, T_c$ is χ -bounded. It is a direct consequence of Ramsey's Theorem that the conjecture holds for stars and it is not hard (see Gyárfás [7]) to show that $\text{Forb}(P)$ is χ -bounded if P is a path. A radius two tree is a tree which contains a vertex r such that every other vertex can be reached from r by a path with at most two edges. The first difficult result in this area is due to Gyárfás, Szemerédi and Tuza [11], who showed that for every radius two tree T , there exists a constant c_T such that $\chi(G) \leq c_T$, for every graph $G \in \text{Forb}(T)$ with $\omega(G) = 2$. Ten years later Kierstead and Penrice [24] generalized their argument to prove that $\text{Forb}(T)$ is χ -bounded for any radius two tree. The general conjecture is still open.

Let P_n be the path on n vertices. An old result of Chvátal shows that, for any on-line graph $G^{\ll} \in \text{Forb}(P_4)$, First-Fit uses at most $\omega(G)$ colors to color G . Gyárfás and Lehel [9] made an unexpected breakthrough when they proved that the class $\text{Forb}(P_5)$ is on-line χ -bounded. They also pointed out that $\text{Forb}(P_6)$ is not on-line χ -bounded. Their on-line algorithm was quite complicated and their binding function was super exponential. They asked for a better binding function and whether $\text{Forb}(P_5)$ was First-Fit χ -bounded. Both questions were answered by Kierstead, Penrice, and Trotter.

Theorem 5.2 (Kierstead, Penrice, and Trotter [25]) *There exists an on-line algorithm $\mathbf{A}$ such that $\mathbf{A}$ colors any on-line graph $G^{\ll} \in \text{Forb}(P_5)$ with at most $(4^{\omega(G)} - 1)/3$ colors.*

Theorem 5.3 (Kierstead, Penrice, and Trotter [25]) *For any tree T , the class $\text{Forb}(T)$ is First-Fit χ -bounded iff T does not contain $K_2 + K_1 + K_1$, the graph on four vertices with one edge, as an induced subgraph.*

Let S be the radius two tree S formed by identifying the first vertices of three copies of P_3 (see Figure 1). Spurred on by the observation of Gyárfás, that the class of co-comparability graphs is contained in $\text{Forb}(S)$, Kierstead, Penrice, and Trotter proved:

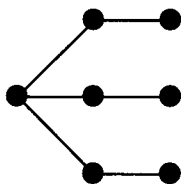


Figure 1

Theorem 5.4 (Kierstead, Penrice and Trotter [26]) *For any tree T , $\text{Forb}(T)$ is on-line χ -bounded iff T has radius at most two.*

Thus Question 3.2 has an affirmative answer. The general proof is much too long to present here; however, we will illustrate the main ideas relating to on-line algorithms by proving the special case, which answers Schmerl's question:

Corollary 5.5 *Forb(S) is on-line χ -bounded.*

Proof. We first prove that Forb(S) is (off-line) χ -bounded. Then we use the proof as a basis for constructing an on-line algorithm. Let $R(\omega, \alpha)$ be the Ramsey function such that for any graph G either $\omega(G) \geq \omega$ or $\alpha(G) \geq \alpha$. Let f be the function on the positive integers defined inductively by:

$$f(1) = 1, \quad \text{and}$$

$$f(\omega) = f(\omega - 1) + \omega + \omega f(\omega - 1) (2(R(\omega, R(\omega + 1, 3)) + 1)^2 + 1).$$

We shall prove by induction on $\omega(G)$ that $\chi(G) \leq f(\omega(G))$ for every graph $G \in \text{Forb}(S)$. The base step, $\omega = 1$, is trivial, so consider a graph $G \in \text{Forb}(S)$ with $1 < \omega(G) = \omega$.

Partition the vertex set V of G into sets $W_1, \dots, W_t, X$ as follows. Suppose we have constructed W_j for $j < i$. Let G_i be the subgraph of G induced by $Y_i = V - \cup_{j < i} W_j$. If $\omega(G_i) < \omega$, then set $t = i - 1$ and $X = Y_i$. Otherwise choose an ω -clique Q_i in G_i . Let $W_i = Q_i \cup N_i$, where $N_i = (N(Q_i) - Q_i) \cap Y_i$. The ω -cliques Q_i are called *templates* and the sequence $W_1, \dots, W_t, X$ is called a *template sequence*. By the induction hypothesis, we can color X with $f(\omega - 1)$ colors. These colors will not be used on any of the vertices of $V - X$. Thus it suffices to show that

$$\chi(G - X) \leq \omega + \omega(f(\omega - 1) (2(R(\omega, R(\omega + 1, 3)) + 1)^2 + 1)).$$

Each W_i can be colored with $\omega f(\omega - 1) + \omega$ colors: First use a distinct color for each element of Q_i . Let $Q_i = \{x_{i1}, \dots, x_{i\omega}\}$. For $j \in \{1, \dots, \omega\}$ let

$$N_{ij} = \{v \in (N(Q_i) - Q_i) \cap Y_i : v \sim x_{ij} \text{ and not } v \sim x_{ik}, \text{ for } k < j\}.$$

Then each N_{ij} is $f(\omega - 1)$ -colorable by the induction hypothesis. Thus, using separate collections of colors for each of the sets $Q_i, N_{i1}, \dots, N_{i\omega}$, we can color W_i with $\omega f(\omega - 1) + \omega$ colors.

Note that no vertex in Q_i is adjacent to any vertex in Q_j , if $i \neq j$. Thus we can use the same set of ω colors to color all the templates Q_i . However, if we try to color each of the $N_i = W_i - Q_i$ with the same set of $f(\omega - 1)$ colors, two adjacent vertices $v_i \in N_i$ and $v_j \in N_j$ may be assigned the same color. To avoid this problem each vertex of $N = V - (X \cup \cup_{1 \leq i \leq t} Q_i)$ will

be assigned a two coordinate color. The first coordinate, assigned as above and called the *local* coordinate, will ensure that two adjacent vertices in the same N_i are assigned different colors. The second coordinate, called the *global* coordinate, will take care of the problem of adjacency between vertices in different N_i . Since t is unbounded, we cannot simply use disjoint sets of colors for the different N_i .

We shall need the following lemma, which is the only place in the proof of the theorem that we use the hypothesis that $G \in \text{Forb}(S)$.

Lemma 5.6 *There exists a function $d(s, \omega)$ such that for every vertex v in any graph $G \in \text{Forb}(S)$ with template sequence $W_1, \dots, W_t, X$, v is connected to vertices in at most $d(s, \omega(G))$ templates by paths with at most s edges.*

Proof. Let $d(s, \omega)$ be the function defined recursively by

$$d(1, \omega) = 2, \quad \text{and}$$

$$d(s, \omega) = (R(\omega, R(\omega + 1, 3)) + 1) d(s - 1, \omega).$$

We argue by induction on s . The base step $s = 1$ follows from the fact $G \in \text{Forb}(S)$: Suppose $v \sim q_i$, where $q_i \in Q_i$, for $i \in \{j_1 < j_2 < j_3\}$. Since each Q_i is a maximum clique there exist vertices $y_i \in Q_i$ such that not $v \sim y_i$, for $i \in \{j_1 < j_2 < j_3\}$. But then $\{v, q_i, y_i : i \in \{j_1 < j_2 < j_3\}\}$ induces S in G , which is a contradiction.

For the induction step, suppose that a vertex v is connected to vertices in $d + 1$ distinct templates by paths with at most s edges, where $d = d(s, \omega)$. Choose a minimal set of vertices $F \subset N(v)$ such that each of these templates either

- (1) contains a vertex which is connected to v by a path with at most $s - 1$ edges, or
- (2) not (1), and contains a vertex which is connected to some vertex in F by a path with exactly $s - 1$ edges.

By the induction hypothesis,

$$|F| + 1 \geq \frac{(d + 1)}{d(s - 1, \omega)} \geq R(\omega, R(\omega + 1, 3)) + 1.$$

Using Ramsey's Theorem, and the fact that every vertex in F is adjacent to v , there exists an independent subset

$$F_0 = \{v_1, \dots, v_p\} \subset F$$

with cardinality $p = R(\omega + 1, 3)$. For each $v_i \in F_0$ there exists a template T_i such that v_i is the only vertex in $F \cup \{v\}$ to which any vertex of T_i is connected by a path R_i with exactly $s - 1$ edges. Say $v \sim v_i \sim y_i$ in R_i . Note that neither $v \sim y_j$ nor $v_i \sim y_j$, if $i \neq j$. Using Ramsey's Theorem again, there exist $j_1 < j_2 < j_3$ such that $\{y_i : i \in \{j_1, j_2, j_3\}\}$ is an independent set. But then $\{v, v_i, y_i : i \in j_1, j_2, j_3\}$ induces S in G , which is a contradiction. $\square$

In order to assign a global color to the vertices in N , we construct auxiliary graphs $A_1, \dots, A_\omega$. The vertex set of A_j is the set of templates $\{Q_1, \dots, Q_t\}$. Two templates Q_x and Q_y are adjacent if there is a path from x_{xj} to x_{yj} with at most three edges. By Lemma 5.6, the maximum degree of A_j is at most $2(R(\omega, R(\omega + 1, 3)) + 1)^2$. Thus A_j can be colored with

$$2(R(\omega, R(\omega + 1, 3)) + 1)^2 + 1$$

colors. The global coordinate of a vertex $v \in N_{ij}$ is the color of Q_i in A_j . To see that this gives a proper coloring, consider two adjacent vertices x and y in N . If x and y have the same local coordinate, there exist indices i, j and k such that $x \in N_{ij}$ and $y \in N_{kj}$. But then Q_i is adjacent to Q_k in A_j , so x and y are assigned different global coordinates. This completes the proof of the off-line case.

We still must show that there exists a function g and an on-line algorithm $\mathbf{A}$ such that $\chi_{\mathbf{A}}(G) \leq g(\omega(G))$, for every graph $G \in \text{Forb}(S)$. It suffices to show by induction on ω that there exist, uniformly, on-line algorithms $\mathbf{A}_\omega$, for $\omega = 1, 2, \dots$ and a function $h(\omega)$ such that $\chi_{\mathbf{A}}(G) \leq h(\omega)$, for every graph $G \in \text{Forb}(S)$ such that $\omega(G) = \omega$: First guess that $\omega = 1$. If a 2-clique is found, guess that $\omega(G) = 2$ and start using $\mathbf{A}_2$ with a new set of colors, etc.. Then $g(\omega) = \sum_{j \leq \omega} h(j)$. The base step is trivial, so consider the induction step.

The major problem in developing an on-line algorithm from the proof of the off-line case is that we cannot possibly construct the template sequence on-line. However, it will suffice to (roughly) maintain an r.e. sequence of templates Q_i and a co-r.e. set X . The key idea is to consider the neighbors of vertices in templates. A minor problem will be that we cannot

maintain the auxiliary graphs on-line. Our on-line algorithm $\mathbf{A} = \mathbf{A}_\omega$ will maintain a list of templates $Q_1, \dots, Q_{t(s)}$, where

$$Q_i = \{x_{i,1} \ll \dots \ll x_{i,\omega}\}.$$

A template Q_i will enter the end of the list at the time $x_{i,\omega}$ is presented. Once a template has entered the list it will not change position or leave the list. When a new vertex v_s is presented, $\mathbf{A}$ will assign v_s to exactly one of the sets N, L, D , or H . The sets N and D are more finely partitioned as:

$$N = \bigcup \{N_{i,j} : i \in \{1, \dots, t(s)\}, j \in \{1, \dots, \omega\}\}, \quad \text{and}$$

$$D = \bigcup \{D_i : i \in \{1, \dots, t(s)\}\}.$$

Then each of the sets of vertices N, L, D , and H , will be colored with disjoint sets of colors.

- (N) If v_s is adjacent to some vertex in some template in the current template list, let i be the least index such that $v_s \sim x_{i,k}$, for some k , and let j be the least such k . Put v_s in $N_{i,j}$.
- (L) Otherwise, if v_s is in an ω -clique Q in $G - (N \bigcup \bigcup_{1 \leq i \leq t(s)} Q_i)$, then add $Q = Q_{t(s)+1}$ to the template list and put v_s in L .
- (D) Otherwise, if v_s is connected to some vertex in some template in the template list by a path with two edges, let i be the *largest* index such that, for some j , v_s is connected to some $x_{i,j}$. Put v_s in D_i .
- (H) Otherwise put v_s in H .

Clearly $\omega(H) < \omega$. Thus the on-line algorithm $\mathbf{A}$ can use the on-line algorithm $\mathbf{A}_{\omega-1}$ to color the vertices of H with one set of $h(\omega - 1)$ colors. Also L is an independent set, so we can use one special color to color L . Thus it remains to color the vertices of N and D . As in the off-line proof, each vertex in N , and also D , will be assigned a two coordinate color. The local coordinate ensures that two adjacent vertices in

$$N_i = \bigcup_{1 \leq j \leq \omega} N_{i,j}$$

or D_i are assigned different colors, while the global coordinate insures that two adjacent vertices with the same local color are assigned different colors.

We first consider the local coordinate. For N the local coordinate is assigned as in the off-line case, using the on-line algorithm $\mathbf{A}_{\omega-1}$. In order to use $\mathbf{A}_{\omega-1}$ to assign a local coordinate to the vertices of D_i , we must first show that $\omega(D_i) < \omega$. Suppose $Q = \{q_1 \ll \cdots \ll q_\omega\}$ is an ω -clique in D_i and consider the situation at the time q_ω was presented. Since q_ω was added to D_i , instead of N , q_ω is not adjacent to any vertex in any template in the template list. Since q_ω was not added to L , some $q \in Q$ must be adjacent to some vertex in a template Q_j . Since q is not in N , Q_j must have been added to the template list after q was presented and thus $i < j$. But then q_ω would have been assigned to D_j . We conclude that $\omega(D_i) < \omega$ for all i , and assign the local coordinate to each vertex in D using $\mathbf{A}_{\omega-1}$.

It remains to determine the global coordinate. First consider the vertices of N . If we could color the vertices of the auxiliary graphs A_j on-line, we would be done. However this is not possible since the auxiliary graphs are not presented on-line. Two templates Q_x and Q_y may start out being non-adjacent, but when a new vertex of $G^{\ll}$ is presented they may suddenly become adjacent. On the other hand, the degree of a template in A_j can only increase $d(3, \omega)$ times. The on-line algorithm $\mathbf{A}$ will maintain a two coordinate, $(d(3, \omega) + 1)^2$ -coloring of A_j such that the first coordinate of a template is the current degree of the template in A_j , the second coordinate ensures that two templates which are adjacent in A_j and have the same degree in A_j are assigned different colors, and the second coordinate of a color assigned to a template will only change when the degree of the template changes. The global coordinate of a vertex in N_{ij} will be the color assigned to Q_i in A_j by $\mathbf{A}$ at the time the vertex is presented.

To assign the global coordinate to a vertex in D , we define another auxiliary graph A on the templates, where two templates Q_x and Q_y are adjacent iff there is a path from a vertex in Q_x to a vertex in Q_y with at most six edges. Since all vertices in a template are adjacent, the maximum degree of A is bounded by $d(7, \omega)$. Thus A is $(d(7, \omega) + 1)$ -colorable and as above we need only

$$(d(7, \omega) + 1)^2$$

colors for the global coordinate of vertices in D . Thus $h(\omega)$ is defined recursively by $h(1) = 1$ and

$$\begin{aligned} h(\omega) = & \omega h(\omega - 1) (d(3, \omega) + 1)^2 + 1 \\ & + h(\omega - 1) (d(7, \omega) + 1)^2 + h(\omega - 1). \end{aligned} \quad \square$$

6 An application

In this section we present one application of on-line coloring to the theory of polynomial time computation. The following storage problem was shown to be NP-complete by Stockmeyer [37].

Dynamic Storage Allocation (DSA).

Instance: Set A of items to be stored, each $a \in A$ having a size $s(a) \in \mathbb{Z}^+$, an arrival time $r(a) \in \mathbb{Z}_0^+$, and a departure time $d(a) \in \mathbb{Z}^+$, and a positive storage size D .

Question: Is there a feasible allocation of storage for A , i.e., a function $\sigma : A \rightarrow \{1, 2, \dots, D\}$ such that for every $a \in A$ the allocated storage interval

$$I(a) = [\sigma(a), \sigma(a) + 1, \dots, \sigma(a) + s(a) - 1]$$

is contained in $[1, D]$ and such that, for all $a, a' \in A$, if $I(a) \cap I(a')$ is nonempty then either $d(a) \leq r(a')$ or $d(a') \leq r(a)$?

The natural next step is to investigate polynomial time approximation algorithms for the optimization version of DSA. First we rephrase DSA as a weighted interval graph coloring problem. A *weighted graph* $G^* = (V, E, w)$ is a graph $G = (V, E)$ together with a *weight* function w which assigns a positive integer to each vertex of G . The *weight* $w(S)$ of a set of vertices $S \subset V$ is the sum of the weights of the vertices in the set. The *weighted clique size* of G^* , denoted by $\omega^*(G^*)$, is the maximum weight of a clique in G . A *weighted t -coloring* of G^* is a function c on V such that:

- (0) $c(v)$ is an interval of the integers $\{1, \dots, t\}$,
- (1) if $v \sim u$, then $c(v) \cap c(u) = \emptyset$, and
- (2) $|c(v)| = w(v)$.

The *weighted chromatic number* $\chi^*(G^*)$ of a weighted graph G^* is the least t such that G^* has a weighted t -coloring. If $\mathbf{A}$ is an algorithm (not necessarily on-line) which produces a weighted coloring, let $\chi_{\mathbf{A}}^*(G^*)$ denote the number of colors that $\mathbf{A}$ uses on G .

Observe that DSA is equivalent to determining whether a weighted interval graph has weighted chromatic number at most D : The vertices of the interval graph correspond to the objects to be stored in DSA. These vertices are represented by the time interval for which the corresponding objects must be stored. The weight of a vertex is the size of the corresponding object. The vertices are colored by the interval of storage positions that the corresponding object occupies.

Let $G^* = (V, E, w)$ be a weighted interval graph. Clearly $\omega^*(G^*) \leq \chi^*(G^*)$. Thus if we can find a polynomial time algorithm $\mathbf{A}$ such that for all weighted interval graphs G^* , $\chi_{\mathbf{A}}^*(G^*) \leq c\omega^*(G^*)$ we will know $\chi^*(G)$ within a factor of c . In this case we say that $\mathbf{A}$ has *constant performance ratio* c . Woodall [42], and later Chrobak and Slusarek [3], suggested the following approach.

- (1) Suppose that every vertex has weight a power of two. This contributes a factor of at most two to our constants.
- (2) Let $\ll^*$ order the vertices by decreasing weight.
- (3) Form an (unweighted) on-line interval graph $H \ll$ from G^* and $\ll$ by replacing every vertex v of G by $w(v) \ll$ -consecutive copies of v , all represented by the same interval as v .
- (4) Color this on-line interval graph using First-Fit.
- (5) Obtain a weighted coloring of G^* by coloring each vertex v with the set of colors assigned to each copy of v .

It is routine to verify that this algorithm produces a weighted t -coloring, with $t \leq 2\chi_{FF}(\omega^*(G^*))$, where $\chi_{FF}(\omega)$ is the maximum number of colors used by First-Fit to color an interval graph with clique size ω . The key point is that because the original vertices are ordered by decreasing size and their sizes are powers of two, First-Fit will (*) assign adjacent colors to adjacent copies of the same vertex. Note the on-line nature of this new problem. The interval graph coloring algorithm First-Fit has no control over the presentation, which is determined solely by the original weights of the vertices.

The following theorem, which answers a question of Woodall [42], shows that DSA has a polynomial time approximation algorithm with a performance ratio of 80.

Theorem 6.1 (Kierstead [18]) *For every on-line interval graph $G^{\ll}$, $\chi_{FF}(G^{\ll}) \leq 40 \omega(G)$.*

Recently Kierstead and Qin [27] improved the constant to 25.72. Chrobak and Slusarek used a very clever induction to prove:

Theorem 6.2 (Chrobak and Slusarek [3]) *There exists an on-line interval graph $G^{\ll}$ such that $4.4\omega(G^{\ll}) \leq \chi_{FF}(G^{\ll})$.*

Thus the technique outlined above cannot achieve a better performance ratio than 8.8 for DSA. However, it turns out that a slight modification of the optimal on-line algorithm of Kierstead and Trotter for coloring on-line interval graphs also satisfies (*) above. This yields:

Theorem 6.3 (Kierstead [19]) *There exists a polynomial-time approximation algorithm for DSA with a performance ratio of 6.*

Acknowledgments

I would like to thank Sarmad Abbasi, William Gasarch, and Katalin Kolossa for their useful comments and proofreading.

References

- [1] D. R. Bean, Effective coloration, *J. Symbolic Logic*, **41** (1976) 469–480.
- [2] R. L. Brooks, On colouring the nodes of a network, *Proc. Cambridge Philos. Soc.*, **37** (1941) 194–197, [also in: *Classic Papers in Combinatorics*, I. Gessel and G.-C. Rota (eds.), (Birkhäuser, Boston, 1987), 118–121].
- [3] M. Chrobak and M. Slusarek, On some packing problems related to Dynamic Storage Allocation, *RAIRO Inform. Théor. Appl.*, **22** (1988) 487–499.
- [4] V. Chvátal, Perfectly ordered graphs, in: *Topics on Perfect Graphs*, C. Berge and V. Chvátal, (eds.), North-Holland Math. Studies, **88**, Ann. Discrete Math., **21** (1984) 63–65.
- [5] P. Erdős and A. Hajnal, On chromatic number of graphs and set-systems, *Acta Math. Acad. Sci. Hungar.*, **17** (1966) 61–99, corr. with B. Rothchild in: On chromatic number of graphs and set-systems, in: (Proc. Cambridge Summer School Math. Logic, Aug. 1–21, 1971), A. R. D. Mathais and H. Rogers, (eds.), *Lecture Notes in Math.*, **337** (1973) 531–538.

- [6] A. Gyárfás, On Ramsey Covering-Numbers, in: *Infinite and Finite Sets*, Vol II, (to P. Erdős on his 60th. birthday, Keszthely, 1973), A. Hajnal, R. Rado and V. T. Sós, (eds.), *Coll. Math. Soc. János Bolyai*, **10** (1975) 801–816.
- [7] A. Gyárfás, Problems from the world surrounding perfect graphs, (*Tanulmányok–MTA Számítástech. Automat. Kutató Int.*, Budapest, **177**), *Zasto. Mat.*, **19** (1985) 413–441.
- [8] A. Gyárfás and J. Lehel, On-line and first-fit coloring of graphs, *J. Graph Theory*, **12** (1988) 217–227.
- [9] A. Gyárfás and J. Lehel, Effective on-line coloring of P_5 -free graphs, *Combinatorica*, **11** (1991) 181–184.
- [10] A. Gyárfás and J. Lehel, First-Fit and on-line chromatic number of families of graphs, in: *Proc. 12th. British Combinatorial Conf.*, July 3–7, 1989, Univ. E. Anglia, Norwich, J. Siemons, (ed.), *Ars Combin.*, **29** (1990) 168–176.
- [11] A. Gyárfás, E. Szemerédi and Zs. Tuza, Induced subtrees in graphs of large chromatic number, *Discrete Math.*, **30** (1980) 235–244.
- [12] S. Irani, Coloring inductive graphs on-line, in: *Foundations of Computer Science*, (Proc. 31st. Annual Sympos. Foundations Comput. Sci., St. Louis, MO, Oct. 22–24, 1990), *FOCS-90*, (IEEE Comput. Soc. Press, Los Alamitos, CA, 1990), Vol. 2, 470–479.
- [13] S. Irani, Coloring inductive graphs on-line, *Algorithmica*, **11** (1994) 53–72.
- [14] D. Kelly and W. T. Trotter Jr., Dimension theory for ordered sets, in: *Ordered Sets*, (Proc. NATO Adv. Study Inst., Banff, Alta, Aug. 28 – Sept. 12, 1981), I. Rival (ed.), *NATO Adv. Study Inst. Ser. C: Math. Phys. Sci.*, (Reidel, Dordrecht and Boston), **83** (1982) 171–212.
- [15] H. A. Kierstead, An effective version of Dilworth’s theorem, *Trans. Amer. Math. Soc.*, **268** (1981) 63–77.
- [16] H. A. Kierstead, Recursive colorings of highly recursive graphs, *Canad. J. Math.*, **33** (1981) 1279–1290.
- [17] H. A. Kierstead, An effective version of Hall’s theorem, *Proc. Amer. Math. Soc.*, **88** (1983) 124–128.

- [18] H. A. Kierstead, The linearity of first-fit coloring of interval graphs, *SIAM J. Discrete Math.*, **1** (1988) 526–530.
- [19] H. A. Kierstead, A polynomial time approximation algorithm for dynamic storage allocation, (Special issue: Combinatorics of Ordered Sets, papers from the 3rd. Oberwolfach Conf., April, 1988), M. Aigner and R. Wille, (eds.), *Discrete Math.*, **88** (1991) 231–237.
- [20] H. A. Kierstead, Long stars specify χ -bounded classes, in: *Sets, Graphs, and Numbers, a birthday salute to V. T. Sós and A. Hajnal*, (Jan. 20–26, 1991, Budapest, Hungary), G. Halász, L. Lovász, D. Miklós and T. Szönyi, (eds.), *Coll. Math. Soc. János Bolyai*, **60** (1992) 421–428.
- [21] H. A. Kierstead, On line coloring k -colorable graphs, *Israel J. Math.*, (to appear).
- [22] H. A. Kierstead, G. McNulty and W. T. Trotter Jr., A theory of recursive dimension for ordered sets, *Order*, **1** (1984) 67–82.
- [23] H. A. Kierstead and S. G. Penrice, Recent results on a conjecture of Gyárfás, in: *Proc. 21st. S. E. Conf. Combinatorics, Graph Theory and Computing*, (Boca Raton, FL, Feb. 12–16, 1990), F. Hoffman, R. C. Mullin, R. G. Staton and K. B. Reid, (eds.), *Congr. Numer.*, **79** (1990) 182–186.
- [24] H. A. Kierstead and S. G. Penrice, Radius two trees specify χ -bounded classes, *J. Graph Theory*, **18** (1994) 119–129.
- [25] H. A. Kierstead, S. G. Penrice and W. T. Trotter Jr., On-line coloring and recursive graph theory, *SIAM J. Discrete Math.*, **7** (1994) 72–89.
- [26] H. A. Kierstead, S. G. Penrice, and W. T. Trotter Jr., On-line and First-Fit coloring of graphs that do not induce P_5 , *SIAM J. Discrete Math.*, **8** (1995) 485–498.
- [27] H. A. Kierstead and J. Qin, Coloring interval graphs with First-Fit, (Special issue: Combinatorics of Ordered Sets, papers from the 4th. Oberwolfach Conf., 1991), M. Aigner and R. Wille, (eds.), *Discrete Math.*, **144** (1995) 47–57.
- [28] H. A. Kierstead and W. T. Trotter Jr., An extremal problem in recursive combinatorics, in: *Combinatorics, Graph Theory and Computing*, Vol. 2, (Proc. 12th. S. E. Conf., Baton Rouge, LA, 1981, *Congr. Numer.*, **33** (1981) 143–153.

- [29] H. A. Kierstead and W. T. Trotter, Colorful induced subgraphs, (Special issue: to mark the centennial of Julius Petersen's "Die Theorie der regulären graphs", Part 2), *Discrete Math.*, **101** (1992) 165–169.
- [30] L. Lovász, M. Saks and W. T. Trotter Jr., An on-line graph coloring algorithm with sublinear performance ratio, (Special issue: Graph Theory and Combinatorics, Proc. Cambridge Combinatorial Conf. in honour of P. Erdős, Mar. 21–25, 1988), B. Bollobás, (ed.), *Discrete Math.*, **75** (1989) 319–325.
- [31] A. B. Manaster and J. G. Rosenstein, Effective matchmaking (recursion theoretical aspects of a theorem of Philip Hall, *Proc. London Math. Soc.*, Ser. 3, **25** (1972) 615–654.
- [32] S. G. Penrice, Techniques for coloring graphs with forbidden trees, Ph.D. Thesis, Arizona State Univ., Tempe, AZ, (1991).
- [33] S. G. Penrice, On-line transitive orientation, (preprint, 1992).
- [34] J. H. Schmerl, (private communication, 1978).
- [35] J. H. Schmerl, Recursive colorings of graphs, *Canad. J. Math.*, **32** (1980) 821–830.
- [36] J. H. Schmerl, The effective version of Brooks' theorem, *Canad. J. Math.*, **34** (1982) 1036–1046.
- [37] L. J. Stockmeyer, (private communication, 1976).
- [38] D. P. Sumner, Subtrees of a graph and the chromatic number, in: *The Theory and Applications of Graphs*, G. Chartrand, (ed.) (John Wiley and Sons, NY, 1981), 557–576.
- [39] M. Szegedy, (private communication, 1986).
- [40] E. Szemerédi and W. T. Trotter Jr., (private communication, 1981).
- [41] S. Vishwanathan, Randomized online graph coloring, *J. Algorithms*, **13** (1992) 657–669.
- [42] D. R. Woodall, Problem No. 4, Combinatorics, in: (Proc. British Combinatorial Conf. 1973), T. P. McDonough and V. C. Marvon, (eds.), London Math. Soc. Lecture Note Ser., **13** (1974) 202.

- [43] A. C. Yao, Probabilistic Computations: Toward a unified measure of complexity, in: Foundations of Computer Science, (Proc. 18th. Annual Sympos. Foundations Comput. Sci., Providence, RI, Oct. 31 – Nov. 2, 1977), FOCS-77, (IEEE Comput. Soc. Press, Washington, DC, 1977), 222–227.

H. A. Kierstead
Department of Mathematics,
Arizona State University,
Tempe, AZ 85287-1804, USA
e-mail: kierstead@math.la.asu.edu

This Page Intentionally Left Blank

Chapter 19

Polynomial-Time Computability in Analysis

K. Ko*

Contents

Introduction

1. Recursive analysis
2. Polynomial-time complexity theory of real functions
3. Other approaches

References

Introduction

Research in computational complexity of the past thirty years has clearly established polynomial-time complexity theory (or, the theory of NP-completeness) as the new foundation of algorithms. Recursion theory identifies the intuitive notion of effective computability with the formal notion of recursiveness, and hence provides a formal setting for us to study what problems are computable and what are not computable. Polynomial-time complexity theory identifies the intuitive notion of feasible computability with the formal notion of polynomial-time computability, and allows us to formally study what problems are feasible and what are infeasible. This new theory ties closely the tools in abstract mathematics with practical computational problems. It has made strong impact on almost every area of discrete computation, including graph theory, combinatorial optimization, computational geometry, computational number theory and modern cryptography (see [16]).

*Research supported in part by NSF grant CCR 9121472.

The influence on numerical computation came later but is definitely not weaker. Due to the continuous nature of numerical computation, the discrete polynomial-time complexity theory cannot be applied directly to numerical problems. Extra effort has to be made to formulate formal computational models of continuous functions to allow us to apply discrete complexity theory to numerical problems. Different approaches aiming at different areas of numerical problems may adopt different computational models, but it is universal to identify the notion of polynomial-time computability with the notion of feasible computability.

In this paper, we present a short survey on one of the polynomial-time complexity theories of numerical computation based on the model of recursive analysis. We will call this the *theory of polynomial-time analysis*. In this theory, a real number x is represented by a sequence of rational numbers that converges to x in a predefined rate of convergence; that is, it is represented as a *type-1 function* satisfying the convergence condition. A real function f is then represented by a *type-2 function* satisfying some continuity condition. The advantage of using this model of computation is that it is compatible with the model of discrete complexity theory, and so the complexity of numerical problems may be studied and compared with discrete problems. Using this model, we are able to classify the computational complexity of numerical problems in terms of the complexity classes in the discrete polynomial-time complexity theory such as P, NP and PSPACE. This approach to polynomial-time complexity theory of numerical computation has been first introduced by Ko and Friedman [34], and further established since then in [11, 15, 20, 21, 29, 30, 32, 31, 33, 35, 50].

This paper is organized as follows. First we introduce the computational model and some computability results in recursive analysis, including some recent results by Pour-El and Richards [54, 55, 56]. Next we add the complexity measures to the computational model of recursive analysis to form the model for polynomial-time analysis. Then, a hierarchical classification of the computational complexity of numerical problems is presented in terms of the relations among discrete complexity classes. The problems to be considered including maximization, root-finding, integration, ordinary differential equations, integral equations and measures of two-dimensional regions. We will not include any formal proofs but will give precise definitions and statements of theorems. Finally we discuss different approaches to the polynomial-time complexity theory of real functions, including the works of Blum et al. [7] and Traub et al. [62].

1 Recursive analysis

In this section, we give a short summary of some recent results in recursive analysis. We do not intend to present a complete survey on recursive analysis. Instead, we only include some results to motivate the study of these problems in the context of polynomial-time complexity theory of analysis. The reader is referred to Pour-El and Richards [56] and Weihrauch [65, 66] for more complete treatments. Other interesting results include Metakides and Nerode [39] and Metakides, Nerode and Shore [40].

1.1 Computable real functions

As is well known, there are two different approaches in recursive analysis. The first approach studies computability of real functions in the context of classical real analysis. Nonrecursive objects, as well as nonconstructive arguments, are allowed. A computable real function is defined on all real numbers, though most of them are noncomputable [56]. The second approach is more constructive. It only studies recursive objects and only uses constructive logic. A computable real function is, thus, defined only on computable real numbers [1, 58].¹ We will only discuss recursive analysis in the first approach.

The model of computation in recursive analysis is based on the Turing machine (TM) model.² Turing machines were designed to deal with discrete objects, more specifically, finite strings of symbols. Therefore, when we deal with real numbers and real functions, we must first establish a representation system in which real numbers and real functions are represented as functions or functionals on finite strings. We first fix the set of rational numbers $\mathbf{Q}$ as the base set of numbers which are representable by finite strings; namely, a rational number p/q , in the reduced form, is represented as s_p/s_q , where s_p and s_q are two strings over $\{+, -, 0, 1\}$ representing integers p and q . Based on the notion of Cauchy sequence representation of real numbers, a real number x is represented by a function $\varphi : \mathbb{N} \rightarrow \mathbf{Q}$ having the property that $\{\varphi(n)\}$ converges to x .

¹A third and most constructive approach is the constructive analysis of [4] and [5] which uses intuitionistic logic and does not restrict itself to the notion of recursiveness, and so is quite different from the above two approaches of recursive analysis. Nevertheless, their work is closely related to recursive analysis; see the discussions in [40].

²For the definition of Turing machines, refer to any standard textbook on theory of computation, e.g., [22].

Definition 1.1 A real number x is *recursive* (or *computable*) if there exist recursive functions $\varphi : \mathbb{N} \rightarrow \mathbb{Q}$ and $t : \mathbb{N} \rightarrow \mathbb{N}$ such that $\lim_{n \rightarrow \infty} t(n) = \infty$ and $|\varphi(n) - x| \leq 1/t(n)$.

In other words, x is recursive if it has a recursive representation φ that converges to x recursively. It turns out that the converging rate could be required to be faster without changing the class of the recursive real numbers. In addition, the other definitions based on the notion of Dedekind cuts and binary expansions are also equivalent [57].

Proposition 1.2 *The following are equivalent:*

- (a) x is a recursive real number.
- (b) There exists a recursive function $\varphi : \mathbb{N} \rightarrow \mathbb{Q}$ such that for all $n \geq 0$, $|\varphi(n) - x| \leq 2^{-n}$.
- (c) The set of rational numbers r that are less than x is a recursive set.
- (d) There exists a recursive function $\varphi : \mathbb{N} \cup \{-1\} \rightarrow \mathbb{N}$, with $\varphi(-1) \in \{1, -1\}$ and $\varphi(i) \in \{0, 1\}$ for all $i > 0$, such that

$$x = \varphi(-1) \cdot \left(\sum_{i=0}^{\infty} \varphi(i) \cdot 2^{-i} \right).$$

Although these representations of real numbers define the same class of recursive real numbers, they do not define the same classes of subrecursive real numbers. Through careful studies, it has been established that the Cauchy sequence representation is the most comprehensive representation system [26, 43, 66]. From the above equivalent definition (b), we say a function φ *binary converges* to x if $|\varphi(n) - x| \leq 2^{-n}$.

Since a real number is represented by a function mapping integers to rationals, a real function is a *functional* mapping integer functions to integer functions, or, as commonly called in recursive function theory, a *type-2 function*. Computable functionals have been studied thoroughly in recursive function theory. A standard machine model for computable functionals is the function-oracle Turing machine.

A *function-oracle Turing machine* is an ordinary Turing machine M equipped with an additional query tape and two additional states: the query state and the answer state. When the machine enters the query state, the

oracle, a function φ , replaces the current string s in the query tape by the string $\varphi(s)$, moves the tape head back to the first cell of the query tape, and puts the machine M in the answer state. When the time complexity is considered, the whole process of querying for the value $\varphi(s)$, i.e., the move from the query state to the answer state, costs only one time unit to the machine. The computation of the function-oracle machine M on input s with oracle φ is written as $M^\varphi(s)$.

Definition 1.3 A real function $f : \mathbf{R} \rightarrow \mathbf{R}$ is *recursive* (or *computable*) if there is a function-oracle TM M such that for each $x \in \mathbf{R}$ and each φ that binary converges to x , the function ψ computed by M with oracle φ (i.e., $\psi(n) = M^\varphi(n)$) binary converges to $f(x)$. We say the function f is *computable on interval* $[a, b]$ if the above condition holds for all $x \in [a, b]$.

Similar to computable real numbers, computable real functions have several different but equivalent formulations [17, 18, 36, 37, 38, 44]. It is easy to check that our definition using oracle TM's is also equivalent to Grzegorzczuk's original definition.

One of the most important properties of a computable real function is that it must be continuous on its domain and the modulus of continuity is computable. Our definition of modulus functions is a little different from that commonly used in numerical analysis.

Definition 1.4 Let $f : [a, b] \rightarrow \mathbf{R}$ be a continuous function on $[a, b]$. Then, a function $m : \mathbb{N} \rightarrow \mathbb{N}$ is said to be a *modulus function* of f on $[a, b]$ if $|x - y| \leq 2^{-m(n)}$ implies $|f(x) - f(y)| \leq 2^{-n}$, for all $x, y \in [a, b]$ and for all $n \geq 0$.

Proposition 1.5 If $f : [a, b] \rightarrow \mathbf{R}$ is computable on $[a, b]$, then f is continuous on $[a, b]$. Furthermore, f has a recursive modulus function m on $[a, b]$.

The above proposition allows us to give an equivalent definition of computable real functions without using the notion of oracle Turing machines.

Proposition 1.6 A real function $f : [a, b] \rightarrow \mathbf{R}$ is computable if and only if f has a recursive modulus function m on $[a, b]$ and there exists a recursive function $\psi : (\mathbf{Q} \cap [a, b]) \times \mathbb{N} \rightarrow \mathbf{Q}$ such that for all $r \in \mathbf{Q} \cap [a, b]$ and all $n \geq 0$, $|\psi(r, n) - f(r)| \leq 2^{-n}$.

For simplicity, we only state the above characterization for functions defined on a closed interval. It is, though, easy to extend to functions defined on the whole real line $\mathbf{R}$. By Proposition 1.6, a real function can be represented by two discrete functions: the modulus function m and the approximation function ψ . Using this representation, we may compute a functional F that maps real functions to real numbers by an oracle TM.

Definition 1.7 A numerical functional F on $D \subseteq C[a, b]$ is *computable* if there exists a two-oracle TM M such that for any function $f \in D$, any oracle functions m and φ , and any input $n \in \mathbb{N}$,

$$|M^{m,\varphi}(n) - F(f)| \leq 2^{-n}$$

provided that the functions m and φ represent f as in Proposition 1.6.

Finally, we note that the notion of recursive real functions can be extended in a straightforward way to functions on $\mathbf{R}^n$ for $n > 1$. We omit the formal definitions.

1.2 Roots

In this section, we consider the computability of roots of recursive real functions f defined on a closed interval $[0, 1]$. If f is a one-to-one function, and $f(0) \cdot f(1) < 0$ then by the intermediate value theorem, there must exist a root $x \in [0, 1]$ such that $f(x) = 0$. The strong form of the recursive version of the intermediate value theorem holds so that the functional mapping a recursive, one-to-one real function to its root is recursive.

Theorem 1.8 *Let f be a recursive, one-to-one function on $[0, 1]$ with $f(0) \cdot f(1) < 0$. Then, there exists a unique recursive real number $x \in [0, 1]$ such that $f(x) = 0$. Furthermore, there exists a two-oracle TM M such that for any oracles m and ψ that represent a one-to-one real function f on $[0, 1]$ (in the sense of Definition 1.7), with $f(0) \cdot f(1) < 0$, and for any input $n > 0$, M outputs a rational number r such that $|r - f^{-1}(0)| \leq 2^{-n}$.*

Note that in the above, the output r is an approximation to the unique root of f , instead of an *approximate root* r' satisfying $|f(r')| \leq 2^{-n}$.

If f is not known to be one-to-one, then the roots of f may not be computable. This result was first proved by Specker [61]. The following

characterization result was from Nerode and Hwang [47]. For simplicity, we only state the results on recursive real functions defined on the closed interval $[0, 1]$.

Definition 1.9 A set $S \subseteq \mathbf{R}$ is *recursively open* if it is empty or if there is a recursive function $\varphi : \mathbb{N} \rightarrow \mathbf{Q}$ such that for each $n \in \mathbb{N}$, $\varphi(2n) < \varphi(2n + 1)$, and $S = \bigcup_{n=0}^{\infty} (\varphi(2n), \varphi(2n + 1))$. A set S is *recursively closed* if the set $\mathbf{R} - S$ is recursively open.

Theorem 1.10 A set $S \subseteq [0, 1]$ is recursively closed if and only if there is a recursive function $f : [0, 1] \rightarrow \mathbf{R}$ such that S contains exactly the roots of f in $[0, 1]$.

Corollary 1.11

- (a) There exists a recursive real function f on $[0, 1]$ which has an uncountable number of roots but none of them is computable.
- (b) If x is an isolated root of a recursive function f in $[0, 1]$, then x is recursive.
- (c) If a recursive real function f has only finitely many roots in $[0, 1]$, then all roots of f are recursive.
- (d) If a recursive real function f has a countably infinite number of roots in $[0, 1]$, then it has an infinite number of recursive roots.

1.3 Ordinary differential equations

In this section, we consider the first-order ordinary differential equation with an initial condition

$$y'(x) = f(x, y(x)), \quad y(0) = 0 \tag{1}$$

that is defined by a recursive real function f on the rectangle $[0, 1] \times [-1, 1]$. Assume that f is recursive and that the above equation has a solution y on $[0, 1]$. Is the solution y computable? The answer, similar to the roots of a recursive real function, depends on the number of solutions y of the equation.

Theorem 1.12 (Pour-El and Richards [53]) *There exists a recursive real function $f : [0, 1] \times [-1, 1] \rightarrow \mathbf{R}$ such that the equation (1) defined by f does not have a computable solution y on $[0, \delta]$ for any $\delta > 0$.*

The equation (1) defined by the function f of the above theorem has uncountably many solutions y on $[0, 1]$, but none of them is computable. On the other hand, if the solution is unique then it must be computable.

Theorem 1.13 (Pour-El and Richards [53]) *Assume that f is computable on $[0, 1] \times [-1, 1]$ and the equation (1) defined by f has a unique solution on $[0, b]$, $0 < b \leq 1$. Then, the solution y of equation (1) is computable on $[0, b]$.*

A function f satisfies the *Lipschitz condition* on a set $E \subseteq \mathbf{R}^2$ if there exists a constant L such that for all $(x, y_1), (x, y_2) \in E$,

$$|f(x, y_1) - f(x, y_2)| \leq L \cdot |y_1 - y_2|.$$

The Lipschitz condition on f on the domain $[0, 1] \times [-1, 1]$ is one of the sufficient conditions for the existence of a unique solution y for equation (1). The recursive version of this existence theorem follows from the above theorem.

Corollary 1.14 *Assume that f is computable and satisfies the Lipschitz condition on $[0, 1] \times [-1, 1]$. Then, the solution y of equation (1) is computable on $[0, 1]$.*

1.4 Computability of linear operators

Many computability questions in recursive analysis may be formulated as the computability question of a linear operator on the Banach space $C[0, 1]$. For instance, both the integration operator that maps a real function f to $g(x) = \int_0^x f(t)dt$ and the differentiation operator that maps a real function f to $g(x) = f'(x)$ are linear operators. Pour-El and Richards [55] showed a very interesting characterization of the computability of a linear operator that states as follows: a linear operator is computable if and only if it is bounded. In this section, we discuss this result and some of its applications.

In their original paper, Pour-El and Richards [55] proved this characterization results on linear operators in a very general form; that is, the result holds on linear operators defined on any Banach space with a computability theory. Here, for the sake of simplicity, we will consider only the space $C[0, 1]$. First we state the recursive version of the Weierstrass approximation theorem. In the following, a sequence $\{\varphi_n\}$ of real-valued polynomials is *computable* if there is a computable function g such that $g(n, m)$ is a finite sequence of rational numbers $(b_0, \dots, b_k)$ such that $|a_i - b_i| \leq 2^{-m}$ for all $i \leq k$, where $\varphi_n(x) = \sum_{i=0}^k a_i x^i$.

Theorem 1.15 (Pour-El and Caldwell [51]). *For every computable real function f defined on $[0, 1]$, there is a computable sequence $\{\varphi_n\}$ of real-valued polynomials such that $|\varphi_n(x) - f(x)| \leq 2^{-n}$, for all $n \geq 0$ and $x \in [0, 1]$.*

From the above recursive version of the Weierstrass approximation theorem, we can assume that each real function f is represented as the limit of a sequence of polynomial functions; or, the sequence $\{x^0, x^1, \dots\}$ is an *effective generation sequence* for functions in $C[0, 1]$. Thus, if a linear operator T on $C[0, 1]$ maps this generating sequence to a computable sequence of real functions, then it is potentially computable.

A linear operator T from $C[0, 1]$ to $C[0, 1]$ is *closed* if for any sequence $\{f_n\}$ of functions in $C[0, 1]$ that converges to f , the sequence $\{T(f_n)\}$ converges to $T(f)$. A linear operator T from $C[0, 1]$ to $C[0, 1]$ is *bounded* if there exists a constant $c > 0$ such that for all f in the domain of T , $\|T(f)\| \leq c\|f\|$, where $\|f\|$ is the maximum of f on $[0, 1]$.

Theorem 1.16 *Let $T : C[0, 1] \rightarrow C[0, 1]$ be a closed linear operator whose domain includes all functions x^k , $k \geq 0$, such that the sequence $\{T(x^k)\}$ is computable. Then, T maps computable real functions to computable real functions if and only if T is bounded.*

This result has many interesting applications. We list some of them below.

Corollary 1.17

- (a) *For any recursive real function $f : [0, 1] \rightarrow \mathbf{R}$, the integration function $g(x) = \int_0^x f(t)dt$ is recursive.*
- (b) (Myhill [45]). *There exists a recursive real function $f : [0, 1] \rightarrow \mathbf{R}$ that has a continuous derivative on $[0, 1]$ but its derivative $f'(x)$ is not computable.*

About the derivatives, a positive result exists that states that if f has a continuous second derivative, then the first derivative f' is computable [52]. Indeed, this follows from the following characterization of the computability of derivatives [34].

Theorem 1.18 *Let $f : [0, 1] \rightarrow \mathbf{R}$ be a recursive real function having a continuous derivative on $[0, 1]$. Then, f' on $[0, 1]$ is computable if and only if f' has a recursive modulus function on $[0, 1]$.*

1.5 Computability of a two-dimensional region

For any Jordan curve Γ on the plane $\mathbf{R}^2$, we say that Γ is computable if there is a recursive function $f : [0, 1] \rightarrow \mathbf{R}^2$ that is one-to-one except $f(0) = f(1)$ such that $\Gamma = f([0, 1])$. In this section, we consider the question of whether the interior $\text{Int}(\Gamma)$ of a computable Jordan curve Γ is computable and whether its measure is a recursive real number. The answer depends on the two-dimensional Lebesgue measure of the curve Γ . Namely, if the two-dimensional measure of the curve Γ itself is zero, then the interior is computable and its measure is recursive; and if the measure of Γ is positive, then the interior is not necessarily computable and its measure could be nonrecursive.

First, we must define the notion of computability of a two-dimensional region S . There are a number of different formulations of this concept, and we will only consider a very general concept called *recursive approximability* which can be used to study the notion of computability of any subset of $\mathbf{R}^2$. Basically, a set $S \subseteq \mathbf{R}^2$ is recursively approximable if we can determine whether a given point $\langle x, y \rangle$ of $\mathbf{R}^2$ belongs to S or not with a predefined error probability. In the following, we write $(\varphi, \psi) \in CF(\langle x, y \rangle)$ to denote the fact that φ and ψ binary converge to x and y , respectively.

Definition 1.19 A set $S \subseteq \mathbf{R}^2$ is *recursively approximable* if there exists an oracle TM M such that, for each $n \geq 1$, the two-dimensional Lebesgue measure of the set

$$E_M(n) = \{ \langle x, y \rangle \in \mathbf{R}^2 : (\exists (\varphi, \psi) \in CF(\langle x, y \rangle)) M^{\varphi, \psi}(n) \neq \chi_S(\langle x, y \rangle) \}$$

is at most 2^{-n} .

In general, if a set $S \subseteq \mathbf{R}^2$ is recursively approximable, then its two-dimensional measure $\mu(S)$ is recursive.

In the following, we say a real number x is a *left r.e.* real number if the set $\{r \in \mathbf{Q} : r < x\}$ is an r.e. set. The following results are from [32] and [35].

Theorem 1.20 Assume that Γ is a computable Jordan curve on the plane $\mathbf{R}^2$.

- (a) If the two-dimensional Lebesgue measure of the curve Γ itself is zero, then its interior $\text{Int}(\Gamma)$ must be recursively approximable.
- (b) If the two-dimensional Lebesgue measure of the curve Γ itself is recursive, then the measure of its interior $\text{Int}(\Gamma)$ must also be recursive.

Theorem 1.21

- (a) Assume that Γ is a computable Jordan curve on the plane $\mathbf{R}^2$. Then, the measure of the interior $\text{Int}(\Gamma)$ of the curve Γ is a left r.e. real number.
- (b) For any left r.e. real number $x > 0$, there is a computable Jordan curve Γ such that its interior $\text{Int}(\Gamma)$ has the measure x .

Note that the Jordan curve Γ of Theorem 1.21 (b) must have a positive, nonrecursive two-dimensional Lebesgue measure. We remark that if a computable Jordan curve Γ has a positive two-dimensional Lebesgue measure $\mu(\Gamma)$ then its Hausdorff dimension must be equal to two and hence the curve Γ is a fractal. (For the notions of fractals and the Hausdorff dimension, see, e.g., [14].) On the other hand, when $\mu(\Gamma)$ is equal to zero, the Hausdorff dimension of $\mu(\Gamma)$ is between one and two. In general, it is not necessarily recursive.

Theorem 1.22 (Ko [33]) *There exists a computable Jordan curve Γ on the plane $\mathbf{R}^2$ whose Hausdorff dimension is a nonrecursive real number between 1 and 2.*

2 Polynomial-time complexity theory of real functions

In recursive analysis, we consider only the notion of computability of a numerical problem. Now we add complexity measures to the computation of real functions and consider the complexity of solving a numerical problem. There are two major differences between the results of recursive analysis and polynomial-time analysis. First, the notion of computability in recursive analysis is a universal criterion. Although the degrees of noncomputability may be of interests for some specific problems, the major issue in recursive analysis is to determine whether a problem is computable or not computable. In polynomial-time analysis, however, the situation is different. First, for a given numerical problem, we are not only interested in the polynomial-time computability or noncomputability of the solution, but also interested in identifying precisely its inherent complexity. Furthermore, for many problems, its polynomial-time computability is hard to prove or

disprove absolutely, and the best we can do is to classify the problem into some well-known complexity classes in discrete complexity theory. Different numerical problems may require different proof techniques to classify their inherent complexity, and yet these different results allow us to compare their complexity in a single, uniform model. Indeed, as we will see, the complexity of different numerical operations forms a hierarchy that is parallel to the hierarchy of the discrete complexity classes.

Second, it is important to point out that certain critical analytic properties seem to affect greatly the inherent complexity of a numerical problem. In the last section, we have seen some results indicating that the computability of a numerical solution often depends not only on the computability of the input function defining the problem but also on some analytic properties of the function. For instance, in Section 1.3, it was shown that the Lipschitz condition on a function f is sufficient to guarantee the computability of the solution of the ordinary differential equation defined by f . In the complexity analysis of these problems, the analytic properties appear to be even more important. Many elementary numerical operations that are easily seen to be computable from the point of view of recursive analysis turn out to have very high complexity unless certain analytic properties are assumed on the input functions. Furthermore, we cannot expect in polynomial-time analysis a simple, clear-cut result like Theorem 1.16 that uses a single analytic property to characterize the computability of many different operations. Instead, it is one of the main issues in polynomial-time analysis to identify the critical analytic properties of the input functions that determine the complexity of the solutions.

One of the similarities between our approach to polynomial-time analysis and recursive analysis is that we do not study the polynomial-time computability of a numerical functional. Instead, we ask a weaker type of questions of whether a numerical functional maps a polynomial-time computable real function to a polynomial-time computable real number (or real function). This weaker approach is necessary since under the worst-case complexity measure, almost all nontrivial numerical functionals are easily provable to be not polynomial-time solvable, based on the simple adversary argument. Such general lower bound results fail to provide insight into the inherent complexity of the underlying numerical functionals. Our weaker approach restricts the domain of the numerical functionals to the set of polynomial-time computable functions. Since the class of polynomial-time computable functions

is a rich, nontrivial class, this approach is not too restrictive and yet allows us to classify the complexity of the numerical functionals more accurately.

2.1 Complexity of real functions

In Section 1.1, we have defined computable real numbers and real functions. In order to define the complexity measures of real numbers and real functions, we need to further fix the representation system of real numbers. Instead of using rational numbers $\mathbf{Q}$ as the base set for approximating real numbers, we choose the set $\mathbf{D}$ of dyadic rational numbers, or, rational numbers with finite binary expansions, as the base set. Set $\mathbf{D}$ has a simple representation system: each dyadic rational number d is naturally represented by a binary string $s = \pm s_n s_{n-1} \cdots s_0 . t_1 t_2 \cdots t_m$ satisfying

$$d = \pm \sum_{i=0}^n s_i \cdot 2^i \pm \sum_{j=1}^m t_j \cdot 2^{-j}.$$

We say that a representation s for a dyadic rational d has the precision m , and write $\text{prec}(s) = m$, if it has m bits to the right of the binary point. We let $\mathbf{D}_n$ denote the class of dyadic rationals d which have a representation of precision $\leq n$; i.e., $\mathbf{D}_n = \{m \cdot 2^{-n} : m \in \mathbf{Z}\}$. The set $\mathbf{D}$ is preferred to the set $\mathbf{Q}$ because its members have simple binary representations and, more importantly, because it is *uniformly dense* in $\mathbf{R}$: the numbers in $\mathbf{D}_n$ are uniformly distributed over the real line.

Based on this representation system, we define the complexity of a real number x to be the complexity of computing a dyadic rational d of precision n that approximates x with an error $\leq 2^{-n}$; i.e., the first n bits of d , to the right of the binary point, are the effective bits of x .

Definition 2.1

- (a) Let t be an integer function. We say that the *time* (or, *space*) *complexity* of a computable real number x is bounded by t if there exists a TM which computes, on each input $n \in \mathbf{N}$, a dyadic rational number d in $t(n)$ moves (or, respectively, using $t(n)$ cells) such that $|d - x| \leq 2^{-n}$.
- (b) A real number x is *polynomial-time computable* if its time complexity is bounded by a polynomial function p .

It is known that all algebraic numbers and many well-known transcendental numbers, including e and π , are polynomial-time computable. Furthermore, the class of polynomial-time computable real numbers forms a real closed field.

We have defined computable real functions using the function-oracle TM's in which input and output real numbers are represented by rational-valued functions that binary converge to them. This definition allows us to define the complexity of real functions naturally.³ For simplicity, we only consider real functions defined on a closed interval $[0, 1]$.

Definition 2.2

- (a) Let $f : [0, 1] \rightarrow \mathbf{R}$ be a computable function. We say that the *time* (or, *space*) *complexity* of f on $[0, 1]$ is bounded by a function $t : \mathbb{N} \rightarrow \mathbb{N}$ if there exists an oracle TM M which computes f (as defined in Definition 1.3 except that the output of $M^\varphi(n)$ is a dyadic rational) such that for all φ that binary converge to a real number $x \in [0, 1]$ and for all $n > 0$, $M^\varphi(n)$ halts in time $t(n)$ (or, respectively, $M^\varphi(n)$ uses at most $t(n)$ cells of workspace⁴).
- (b) A real function $f : [0, 1] \rightarrow \mathbf{R}$ is *polynomial-time computable* if its time complexity is bounded by a polynomial function p .

We have seen in Proposition 1.5 that a computable real function must have a computable modulus function. The following proposition gives a more precise relation between the time complexity and the modulus of continuity of a real function.

Proposition 2.3

- (a) Assume that the time complexity of $f : [0, 1] \rightarrow \mathbf{R}$ is bounded by $t : \mathbb{N} \rightarrow \mathbb{N}$. Then, the function $t(n + 2)$ is a modulus function for f on $[0, 1]$.
- (b) Each polynomial-time computable real function $f : [0, 1] \rightarrow \mathbf{R}$ has a polynomial modulus of continuity.

³As we pointed out in Section 1.1, the time complexity of an oracle TM is defined in such a way that the cost of making a query is only one unit of time.

⁴When we consider the space complexity of computing a real function, we do not include the space of query tape in the space measure. This follows from the convention in discrete complexity theory that the input/output space is not included in the space measure.

This observation gives us a characterization of polynomial-time computable real functions like that of Proposition 1.6.

Corollary 2.4 *A function $f : [0, 1] \rightarrow \mathbf{R}$ is polynomial-time computable if and only if there exist polynomial functions m and q , and a function $\psi : (\mathbf{D} \cap [0, 1]) \times \mathbb{N} \rightarrow \mathbf{D}$ such that*

- (i) *m is a modulus function for f on $[0, 1]$,*
- (ii) *for any $d \in \mathbf{D} \cap [0, 1]$ and all $n \in \mathbb{N}$, $|\psi(d, n) - f(d)| \leq 2^{-n}$, and*
- (iii) *$\psi(d, n)$ is computable in time $q(\text{prec}(d) + n)$.*

2.2 Discrete complexity theory

To prepare for classifying the complexity of numerical problems in terms of discrete complexity classes, we give a short summary of some important notions in discrete complexity theory.

2.2.1 Complexity classes

We first review some important complexity classes. We will only give informal definitions to these complexity classes. More careful discussions of the properties and relations between these complexity classes can be found in, for instance, [3, 16, 29].

First, we consider *language recognition problems* (or, *decision problems*), that is, the problems asking a yes/no answer. Such a problem is represented as a set of strings. The most important complexity classes of language recognition problems include LOGSPACE, P, NP, PSPACE and EXPTIME.

LOGSPACE: the class of sets accepted by deterministic TM's in logarithmic space (i.e., in space $c \log n$ for some constant $c > 0$).

P: the class of sets accepted by deterministic TM's in polynomial time (i.e., in time n^k for some constant $k > 0$).

NP: the class of sets accepted by nondeterministic TM's in polynomial time.

PSPACE: the class of sets accepted by deterministic TM's in polynomial space.

EXPTIME: the class of sets accepted by deterministic TM's in exponential time (i.e., in time 2^{n^k} for some constant $k > 0$).

It is known that $\text{LOGSPACE} \subseteq \text{P} \subseteq \text{NP} \subseteq \text{PSPACE} \subseteq \text{EXPTIME}$. It is easy to prove by diagonalization that $\text{LOGSPACE} \neq \text{PSPACE}$ and $\text{P} \neq \text{EXPTIME}$. Other than these easy separation results, none of the above inclusions is known to be proper. Indeed, these are the major open questions in discrete complexity theory. (It is known that nondeterministic polynomial space defines the same class PSPACE as deterministic polynomial space [59].)

There are many other important complexity classes defined in discrete complexity theory. We will particularly use the following complexity classes:

- UP: the class of sets accepted in polynomial time by nondeterministic TM's that have, for each input, at most one accepting computation path.
- BPP: the class of sets accepted in polynomial time by probabilistic TM's with the error probability bounded by $1/4$.

The class UP is closely related to the notion of one-way functions; namely, weak one-way functions exist if and only if $\text{P} \neq \text{UP}$. The class BPP denotes the class of problems that are feasibly solvable by randomized algorithms. It is known that $\text{P} \subseteq \text{UP} \subseteq \text{NP}$ and $\text{P} \subseteq \text{BPP} \subseteq \text{PSPACE}$. None of these inclusions is known to be proper.

In addition to the language recognition problems, we will also deal with *function evaluation problems* that are represented as functions mapping strings to strings. The main complexity class we are concerned with are the counting class $\#\text{P}$.

- FP: the class of functions computed by deterministic TM's in polynomial time.
- $\#\text{P}$: the class of functions that enumerate the number of accepting computations of polynomial-time nondeterministic TM's.
- FPSpace: the class of functions computed by deterministic TM's in polynomial space.

It is known that $\text{FP} \subseteq \#\text{P} \subseteq \text{FSPACE}$, but the inclusions are not known to be proper.

2.2.2 Reducibility and completeness

The class P is commonly identified with feasibly computable problems. From a complexity point of view, we would like to prove a problem not in P to

demonstrate that it is not feasibly computable. However, a great number of problems arising from different areas of discrete computation turn out to belong to NP or PSPACE and are not known to be in P. In order to give convincing evidence that these problems are not feasible, the notions of reducibility and completeness are applied to these problems.

First, we restrict ourselves to only language recognition problems. A set A is *polynomial-time reducible* to a set B if there exists a polynomial-time computable function φ such that for any input x , $x \in A$ if and only if $\varphi(x) \in B$. A set B is *complete* for a complexity class $\mathcal{C}$ (with respect to the polynomial-time reducibility) if

(a) $B \in \mathcal{C}$, and

(b) for any $A \in \mathcal{C}$, A is polynomial-time reducible to B .

It is easy to see that if A is polynomial-time reducible to B and if B is in P, then A is also in P; it follows that if A is complete for NP (or PSPACE), then $A \in P$ if and only if $P = NP$ (or, respectively, $P = PSPACE$). Garey and Johnson [16] have collected hundreds of problems arising from different areas of research in discrete computation that are proved to be NP- or PSPACE-complete. Proving a new problem A to be NP-complete means that problem A is *equivalent* to all these difficult problems as far as polynomial-time computability is concerned. It also means practically getting a superpolynomial-time *lower bound* for the complexity of problem A in the sense that it is not in P unless $P = NP$.

The notion of completeness can also be extended to problems in P and EXPTIME. Since it is known that $P \neq \text{EXPTIME}$, an EXPTIME-complete problem A is *provably* not feasibly computable. In addition, an EXPTIME-complete problem A is not computable in time NP or space PSPACE, unless $\text{EXPTIME} = \text{NP}$ or $\text{EXPTIME} = \text{PSPACE}$, respectively, which relations are commonly conjectured to be false.

To define completeness for P, we require that the reducibility is computable in log space. More precisely, a set A is *log-space reducible* to a set B if there exists a function φ that is computable in log space⁵ such that for any input x , $x \in A$ if and only if $\varphi(x) \in B$. A set $B \in P$ is *P-complete* if for every set $A \in P$, A is log-space reducible to B .

⁵Following the convention in discrete complexity theory, the space complexity of a function only measures the workspace used by the machine computing the function, not including the input/output space.

The complexity classes below P , including LOGSPACE , NLOGSPACE and NC , are usually identified as the classes of problems that are feasibly solvable by parallel machines.⁶ Therefore, a problem A being P -complete implies that A requires polynomially long sequential computation that is not parallelizable unless the class P collapses to one of the feasible parallel complexity classes.

2.2.3 Complete real functions

In our results on the complexity of numerical problems, we often reduce a discrete complete problem to a numerical problem, using a little weaker type of reducibility. The weaker type of reducibility is necessary as we are dealing with continuous problems and the reducibility is not as straightforward as that for discrete problems; yet the effect is essentially the same.

Definition 2.5 A set $A \subseteq \{0, 1\}^*$ is *polynomial-time reducible* to a real function $f : [0, 1] \rightarrow \mathbf{R}$ if there exist two polynomial-time computable functions $g : \{0, 1\}^* \rightarrow \mathbf{D}$ and $h : \{0, 1\}^* \times \mathbf{D} \rightarrow \{0, 1\}$ and a polynomial function p such that for every $w \in \{0, 1\}^*$,

$$w \in A \iff h(w, e) = 1,$$

where e is any dyadic rational with the property $|e - f(g(w))| \leq 2^{-p(n)}$.

In other words, to determine whether $w \in A$, we can first use g to transfer the instance w for problem A to an instance $d = g(w)$ for problem f , then from an approximate value $e \approx f(d)$, the answer to the question “ $w \in? A$ ” can be found by $h(w, e)$.

Definition 2.6 Let $\mathcal{C}$ be a discrete complexity class. A real function $f : [0, 1] \rightarrow \mathbf{R}$ is *complete* for the class $\mathcal{C}$ if f has a representation (m, ψ) , as defined in Proposition 1.6, such that m is a polynomial function and ψ is computable in $\mathcal{C}$, and if every set $A \in \mathcal{C}$ is polynomial-time reducible to f .

Note that if f is complete for the class $\mathcal{C}$, where $P \subseteq \mathcal{C}$, then f is polynomial-time computable if and only if $P = \mathcal{C}$.

⁶We omit the definitions of the classes NLOGSPACE and NC and their relation to parallel computation. The interested reader is referred to, for instance, [12].

2.3 Maximization

In the next five subsections, we classify the computational complexity of some basic numerical problems in terms of relations among discrete complexity classes. These problems include the maximization problem, the root-finding problem, the integration problem, ordinary differential equations, and integral equations. Additional results can be found in [20, 29, 50]. In the following, we will write $\ell(s)$ to denote the length of a finite string $s \in \{0, 1\}^*$.

2.3.1 Maximization and nondeterminism

The maximization problem in recursive analysis is quite simply solved. Let $f : [0, 1] \rightarrow \mathbf{R}$ be a recursive function. Then the maximum value $m_f = \max\{f(y) : 0 \leq y \leq x\}$ is also recursive; however, the maximum point x_0 such that $f(x_0) = m_f$ is not necessarily recursive (follows from Corollary 1.11).

The maximization problem in polynomial-time analysis is more interesting. First let us review the maximization problem in discrete complexity theory, where many optimization problems were proved to be NP-complete. Consider the following abstract maximization problem. For any function $\varphi : \{0, 1\}^* \times \{0, 1\}^* \rightarrow \{0, 1\}^*$, define

$$\max_{\varphi}(x) = \max\{\varphi(x, y) : \ell(y) = \ell(x)\}.$$

If we know that φ is polynomial-time computable, what is the complexity of computing $\max_{\varphi}$? It is easy to show that $\max_{\varphi}$ is computable in polynomial time if $P = NP$ and, furthermore, there exists a polynomial-time computable φ such that $\max_{\varphi}$ is NP-complete.

The above simple example suggests naturally that we should try to use the notion of nondeterministic computation to characterize the complexity of the maximization problems — even for continuous problems. Indeed, the notion of NP can be used to give a precise characterization of the complexity of the maximization of one- or two-dimensional polynomial-time computable functions. In addition, differentiability of the function does not help reducing the complexity of maximization.

Theorem 2.7 (Friedman [15]). *There exists a polynomial-time computable real function $f : [0, 1] \rightarrow \mathbf{R}$ that is infinitely differentiable (i.e., $f \in C^{\infty}[0, 1]$) such that the function $g(x) = \max\{f(y) : 0 \leq y \leq x\}$ is complete for the class NP.*

Corollary 2.8 *The following are equivalent:*

- (a) $P = NP$.
- (b) *For each polynomial-time computable $f : [0, 1]^2 \rightarrow \mathbf{R}$, the function $g(x) = \max\{f(x, y) : 0 \leq y \leq 1\}$ is polynomial-time computable.*
- (c) *For each polynomial-time computable $f : [0, 1] \rightarrow \mathbf{R}$, the function $h(x) = \max\{f(y) : 0 \leq y \leq x\}$ is polynomial-time computable.*
- (d) *For each polynomial-time computable $f : [0, 1] \rightarrow \mathbf{R}$ that is infinitely differentiable (i.e., $f \in C^\infty[0, 1]$), the function $k(x) = \max\{f(y) : 0 \leq y \leq x\}$ is polynomial-time computable.*

2.3.2 Maximum values and left NP real numbers

Corollary 2.8 shows that computing the maximization function of a polynomial-time computable function is, in the worst case, as difficult as an NP-complete problem. Is the problem of computing the single maximum value $m_f = \max\{f(x) : 0 \leq x \leq 1\}$ as difficult as the problem of computing the maximization function? It turns out to be an interesting question depending on some subtle relations between deterministic and nondeterministic computation.

First we give a characterization of the complexity of the maximum values of polynomial-time computable real functions. Recall that a real number x is represented by a function φ that binary converges to x . For x in $[0, 1]$, such a function φ could also be represented by a set of strings $L_\varphi = \{s \in \{0, 1\}^* : 0.s \leq \varphi(\ell(s))\}$, where $0.s$ denotes the dyadic rational d whose binary expansion is equal to $0.s$. Function φ and set L_φ are polynomially equivalent in the sense that one is computable from the other in polynomial time. Note that L_φ is not equal to the left cut $L = \{d \in \mathbf{D} : d \leq x\}$ unless $\varphi(n) \leq x$ for all $n \geq 1$, but L_φ is very close to the left cut. Such a set L_φ is called a *general left cut* of x .

Definition 2.9 A real number $x \in [0, 1]$ is a left NP real number if it has a general left cut L_φ that is in NP. A real number x is a left NP real number if $x' = x - \lfloor x \rfloor$ is a left NP real number.

Theorem 2.10 (Ko [24]) *A real number x is a left NP real number if and only if there exists a polynomial-time computable real function $f : [0, 1] \rightarrow \mathbf{R}$ such that $x = \max\{f(y) : 0 \leq y \leq 1\}$.*

From the above characterization result, the complexity of maximum values of polynomial-time computable real functions is the same as the complexity of left NP real numbers. To analyze the precise complexity of a left NP real number x , we need to define some new discrete complexity classes. Recall that EXPTIME denotes the class of sets computable in deterministic time 2^{n^k} for some constant k . We define another exponential time complexity classes EXP that contains all sets computable in deterministic time $2^{O(n)}$. We let NEXP denote the class of sets computable in nondeterministic time $2^{O(n)}$. For any complexity class $\mathcal{C}$, we let $\mathcal{C}_1$ denote the class of *tally* sets in $\mathcal{C}$, where a set T is called a tally set if it is a set over a single alphabet $\{0\}$, i.e., if $T \subseteq \{0\}^*$. It is easy to see that $\text{EXP} = \text{NEXP}$ if and only if $\text{P}_1 = \text{NP}_1$ [8].

We note that the structure of a general left cut L_φ of a real number is closely related to the structure of a tally set. The following relations between these classes thus give the best classification of the complexity of left NP real numbers. It implies that if deterministic exponential time is different from nondeterministic exponential time then left NP real numbers are not always polynomial-time computable. Together with Corollary 2.8, it shows that computing a single maximum value has a *lower* complexity than computing the maximum function (of infinitely many maximum values).

Theorem 2.11 (Ko [24]) *In the following, (a) $\implies$ (b) $\implies$ (c) $\iff$ (d).*

- (a) $\text{P} = \text{NP}$.
- (b) *Every left NP real number is polynomial-time computable.*
- (c) $\text{P}_1 = \text{NP}_1$.
- (d) $\text{EXP} = \text{NEXP}$.

2.4 Roots and inverse functions

In this section, we consider the complexity of computing the root and the inverse function of a one-dimensional, one-to-one real function. We will consider only one-to-one real functions. The reason is simple: for functions that are not one-to-one, the roots could be nonrecursive even if the underlying function is polynomial-time computable. The following theorem can be proved from a construction similar to Corollary 1.11.

Theorem 2.12 *There exists a polynomial-time computable real function $f : [0, 1] \rightarrow \mathbf{R}$ such that it has an uncountable number of roots but none of them is computable.*

If we add extra restrictions of analytic properties to the function f , the complexity of its roots becomes reasonably low. However, the conditions are too complicated to be studied here. The interested reader is referred to [29] for the study of the complexity of roots of functions that are not one-to-one.

2.4.1 Inverse modulus of continuity

In Theorem 1.8, we have seen that if a one-to-one function $f : [0, 1] \rightarrow \mathbf{R}$ is recursive, then its inverse function is also recursive. However, the polynomial-time analog of this result does not hold, even if f is infinitely differentiable.

Theorem 2.13 *For any recursive real number $x \in [0, 1]$, there exists a strictly increasing, polynomial-time computable function $f : [0, 1] \rightarrow \mathbf{R}$ that is in $C^\infty[0, 1]$ such that x is the unique root of f in $[0, 1]$.*

Theorem 1.8 was proved by a simple binary search algorithm to find the root. Why does this algorithm fail to find the root in polynomial time? We note that this binary search algorithm can only find an *approximate root* y (such that $|f(y)| \approx 0$) in polynomial time which does not have to be close to the real, unique root x . A sufficient condition for the approximate root to be a good approximation to the real root is that the inverse function f^{-1} has a polynomial modulus function around the root. We say that a real function

$$g : [0, 1] \rightarrow \mathbf{R}$$

has a *polynomially-bounded local modulus* at $z \in [0, 1]$ if g has a polynomial modulus on $[z - \epsilon, z + \epsilon]$ for some $\epsilon > 0$.

Theorem 2.14 (Ko and Friedman [34]) *Assume that f is one-to-one, polynomial-time computable on $[0, 1]$ with the range $[a, b]$, $a < 0 < b$.*

- (a) *If f^{-1} has a polynomially-bounded local modulus at 0 then the root of f in $[0, 1]$ is polynomial-time computable.*
- (b) *If f^{-1} has a polynomial modulus function on $[a, b]$ then f^{-1} is polynomial-time computable on $[a, b]$.*

2.4.2 Fundamental Theorem of Algebra

It is not hard to verify that the inverse function f^{-1} of an analytic, one-to-one, polynomial-time computable function $f : [0, 1] \rightarrow \mathbf{R}$ must have a polynomial modulus function on its domain. Thus the roots of f on $[0, 1]$ are polynomial-time computable. This result can be extended to functions that are not necessarily one-to-one.

Theorem 2.15 (Ko and Friedman [34]) *All roots of an analytic, polynomial-time computable function f on $[0, 1]$ are polynomial-time computable.*

Extending the notion of polynomial-time computable real functions to polynomial-time computable complex-valued functions, we get the following polynomial-time version of the fundamental theorem of algebra. A complex-valued polynomial function is polynomial-time computable if and only if its coefficients are polynomial-time computable complex numbers.

Corollary 2.16 (Polynomial-Time Version of the Fundamental Theorem of Algebra, the weak form). *All roots of a polynomial-time computable complex-valued polynomial function are polynomial-time computable.*

The strong form of this theorem that requires that the mapping from coefficients of a complex-valued polynomial to its roots be polynomial-time computable, though claimed to be true by many numerical analysts, was first *formally* proved true by Schönhage [60] and then improved by Neff [46] showing that this mapping actually is in NC.

Theorem 2.17 (Polynomial-Time Version of the Fundamental Theorem of Algebra, the strong form). *There exists an algorithm in NC (hence in polynomial time) that for any given complex numbers $a_0, a_1, \dots, a_n$, with $a_n = 1$, computes complex numbers $z_1, \dots, z_n$ such that $\sum_{i=0}^n a_i z_j^i = 0$, for $j = 1, \dots, n$.*

2.4.3 Log-space computability of roots

In discrete complexity theory, we often identify polynomial-time computability with sequentially feasible computability, and identify *log-space computability*, or NC computability with *parallelly feasible computability* [12]. In Theorem 2.14, we have established that if the inverse f^{-1} of a one-to-one,

polynomial-time computable function f has a polynomial modulus, then f^{-1} is sequentially feasibly computable. In this section, we investigate the question of whether the inverse functions f^{-1} having such nice properties are parallelly feasibly computable, or, log-space computable.

A real function $f : [0, 1] \rightarrow \mathbf{R}$ is *log-space computable* if it is computable by an oracle TM that uses $O(\log n)$ workspace in its computation on input n , relative to all oracles φ , equivalently, if the functions m and ψ in Proposition 1.6 are log-space computable. A real function

$$g : [0, 1] \rightarrow \mathbf{R}$$

is *complete* for $\mathbf{P}$ if every set $A \in \mathbf{P}$ is *log-space* reducible to g in the sense of Definition 2.5 with the reduction functions g and h computable in log space. The following result shows that even if a function f is log-space computable and satisfies the conditions of Theorem 2.14, the inverse function f^{-1} is not necessarily log-space computable.

Theorem 2.18 (Ko [28]) *There is a log-space computable, one-to-one function $f : [0, 1] \rightarrow \mathbf{R}$ such that f^{-1} has a polynomial modulus but f^{-1} is complete for $\mathbf{P}$.*

Corollary 2.19 *The following are equivalent:*

- (a) $\mathbf{P} = \text{LOGSPACE}$.
- (b) *For each log-space computable, one-to-one function $f : [0, 1] \rightarrow \mathbf{R}$ whose inverse f^{-1} has a polynomial modulus, the inverse f^{-1} is log-space computable.*

For the complexity of a single root, we identify its complexity in terms of $\mathbf{P}_1$ versus LOGSPACE_1 . Recall that $\mathcal{C}_1$ is the class of tally sets in $\mathcal{C}$. The relation between $\mathbf{P}_1$ and LOGSPACE_1 is similar to the relation between exponential time EXP and linear space (cf. discussion before Theorem 2.11).

Theorem 2.20 (Ko [28]) *The following are equivalent:*

- (a) $\mathbf{P}_1 = \text{LOGSPACE}_1$.
- (b) *For each log-space computable, one-to-one function $f : [0, 1] \rightarrow \mathbf{R}$ such that $f(0) < 0 < f(1)$ and that f^{-1} has a polynomial modulus, $x = f^{-1}(0)$ is log-space computable.*

2.5 Integration

We have seen in Corollary 1.17 that integration is a recursive operation. In this section, we show however that the integration of a polynomial-time computable real function f is not necessarily polynomial-time computable, unless the discrete complexity class $\#P$ collapses to FP .

The class $\#P$ is a counting class containing functions that count the number of accepting computations of polynomial-time nondeterministic TM's. To see how this class characterizes the inherent complexity of integration, consider the following discrete counting problem: Assume that a set $A \subseteq \{0,1\}^* \times \{0,1\}^*$ is polynomial-time computable. What is the complexity of computing the function $c_A : \{0,1\}^* \rightarrow \mathbb{N}$ defined by $c_A(x) =$ the number of y such that $\ell(y) = \ell(x)$ and $(x,y) \in A$? It is not hard to see that $FP = \#P$ if and only if for all polynomial-time computable A , the function c_A is polynomial-time computable. We can then convert the function c_A into a continuous integration problem to obtain the following result.

Theorem 2.21 (Friedman [15], Ko [27]) *There exists a polynomial-time computable function $f : [0,1] \rightarrow \mathbb{R}$ such that the function $g(x) = \int_0^x f(t) dt$ is complete for $\#P$.*

Corollary 2.22 *The following are equivalent:*

- (a) $FP = \#P$.
- (b) *For all polynomial-time computable functions $f : [0,1] \rightarrow \mathbb{R}$, the function $g(x) = \int_0^x f(t) dt$ is polynomial-time computable.*

Assume that $\mathcal{F}$ is a complexity class of functions. Let $\mathcal{F}_1$ denote the class of functions f in $\mathcal{F}$ whose domain is exactly $\{0\}^*$. The complexity of computing the integral value over a fixed interval of a polynomial-time computable real function is, similar to that for the maximum values, dependent on the complexity of the class $\#P_1$.

Theorem 2.23 *The following are equivalent:*

- (a) $FP_1 = \#P_1$
- (b) *For all polynomial-time computable functions $f : [0,1] \rightarrow \mathbb{R}$, the value $\int_0^1 f(t) dt$ is a polynomial-time computable real number.*

2.6 Ordinary differential equations

We discussed the computability of the solution of an ordinary differential equation (1)

$$y'(x) = f(x, y(x)), \quad y(0) = 0$$

in Section 1.3. It was proved there that if function f is computable on $[0, 1] \times [-1, 1]$ and if the equation has a unique solution y , then the solution y is computable. The following result, due to Miller [42], however shows that even if the solution y is unique and even if f is polynomial-time computable, the solution y could have arbitrarily high complexity.

Theorem 2.24 *Let a be an arbitrary recursive real number between 0 and 1. Then there is a polynomial-time computable function f defined on $[0, 1] \times [-1, 1]$ such that $y(x) = ax^2$ is the unique solution of equation (1) defined by f .*

The above result suggests that in order to have a feasible solution, some extra analytic condition on the equation (1) must be assumed. One of the most commonly used conditions is the *Lipschitz condition* on f : there exists a constant L such that for all $x \in [0, 1]$ and all $z_1, z_2 \in [-1, 1]$,

$$|f(x, z_1) - f(x, z_2)| \leq L \cdot |z_1 - z_2|.$$

It is known that if f satisfies the Lipschitz condition on $[0, 1] \times [-1, 1]$ then the solution y to equation (1) is unique. Furthermore, for such an equation, a standard successive approximation algorithm such as Euler's method works in polynomial space. The following result follows immediately from this observation and from Corollary 2.22.

Theorem 2.25 *In the following, (a) $\iff$ (b) $\implies$ (c) $\implies$ (d).*

(a) $P = PSPACE$.

(b) $FP = FSPACE$.

(c) *For all polynomial-time computable functions $f : [0, 1] \times [-1, 1] \rightarrow \mathbf{R}$ that satisfy the Lipschitz condition, the solution y of equation (1) is polynomial-time computable.*

(d) $FP = \#P$.

The above result does not precisely characterize the complexity of the ordinary differential equation (1). In the following, we relax the Lipschitz condition a little to give a more precise characterization of the inherent complexity of equation (1). We say that a function f on $[0, 1] \times [-1, 1]$ satisfies the *right Lipschitz condition* on a region $E \subseteq [0, 1] \times [-1, 1]$ if there exists a constant $L > 0$ such that for all $x \in [0, 1]$ and all $z_1, z_2 \in [-1, 1]$, $z_1 < z_2$,

$$f(x, z_2) - f(x, z_1) \leq L \cdot (z_2 - z_1),$$

whenever the line segment connecting (x, z_1) and (x, z_2) lies entirely in region E . We also say that a region $E \subseteq [0, 1] \times [-1, 1]$ *polynomially covers* (the graph of) a function $y : [0, 1] \rightarrow [-1, 1]$ if there is a polynomial q such that for every $k \geq 1$, the set

$$\{(x, z) : 0 \leq x \leq 1 - 2^{-k}, |z - y(x)| \leq 2^{-q(k)}\}$$

is contained in E .

Theorem 2.26 (Ko [25]) *There exists a polynomial-time computable function $f : [0, 1] \times [-1, 1] \rightarrow \mathbf{R}$ such that*

- (i) *equation (1) defined by f has a unique solution y on $[0, 1]$,*
- (ii) *f satisfies the right Lipschitz condition on a region $E \subseteq [0, 1] \times [-1, 1]$ that polynomially covers the solution y , and*
- (iii) *the solution y is complete for PSPACE.*

To prove the above theorem, we first define a discrete initial value problem of the following form. Let $M_k = \{m \in \mathbf{N} : |m| \leq k\}$, and p a polynomial function. We say a function $\varphi : \{0, 1\}^n \times M_{p(n)} \rightarrow M_{p(n)}$ is a *derivative* function if for any string s of length n and integers $m_1, m_2 \in M_{p(n)}$, $\varphi(s, m_1) \leq \varphi(s, m_2)$ whenever $m_1 \leq m_2$. A function $v : \{0, 1\}^n \rightarrow \mathbf{N}$ is a *solution* for the derivative function φ if for all $s \in \{0, 1\}^n - \{1\}^n$, $\varphi(s, v(s)) = v(\text{succ}(s))$, where $\text{succ}(s)$ is the successor of the string s in the lexicographic order on $\{0, 1\}^*$. The discrete initial value problem is, for any given derivative function φ and a given initial value $\varphi(0^n, 0) = m_0$, to compute the *solution* function $v : \{0, 1\}^* \rightarrow \mathbf{N}$ such that $v(0^n) = 0$ and v is a solution of φ . Once such a discrete initial value problem is proved to be complete for PSPACE, we can encode it into a continuous function f to obtain the polynomial space lower bound for equation (1).

Corollary 2.27 *The following are equivalent:*

- (a) *Assume that equation (1) defined by a polynomial-time computable function $f : [0, 1] \times [-1, 1] \rightarrow \mathbf{R}$ has a unique solution y on $[0, 1]$. If f satisfies the right Lipschitz condition on a region $E \subseteq [0, 1] \times [-1, 1]$ that polynomially covers the solution y , then the solution y is polynomial-time computable.*
- (b) $P = PSPACE$.

Similar to the problems about maximization and integration, a single value of the solution y of equation (1) can be characterized by the complexity class $PSPACE_1$.

Theorem 2.28 (Ko [25]) *The following are equivalent:*

- (a) *Assume that equation (1) defined by a polynomial-time computable function $f : [0, 1] \times [-1, 1] \rightarrow \mathbf{R}$ has a unique solution y on $[0, 1]$. If f satisfies the right Lipschitz condition on a region $E \subseteq [0, 1] \times [-1, 1]$ that polynomially covers the solution y , then the value $y(1)$ is a polynomial-time computable real number.*
- (b) $P_1 = PSPACE_1$.

2.7 Integral equations

In this section, we consider the complexity of the solutions of Volterra integral equations of the first kind

$$\int_0^y K(y, s, \varphi(s)) ds = f(y), \quad 0 \leq y \leq 1, \quad (2)$$

and of the second kind

$$\varphi(y) = f(y) + \int_0^y K(y, s, \varphi(s)) ds, \quad 0 \leq y \leq 1, \quad (3)$$

where the function $\varphi : [0, 1] \rightarrow \mathbf{R}$ is to be solved from the given functions $K : [0, 1]^2 \times [-1, 1] \rightarrow \mathbf{R}$ and $f : [0, 1] \rightarrow \mathbf{R}$. While many numerical algorithms for Volterra equations of the second kind have been proposed

and studied (see, e.g., [2, 41, 64]), very few general algorithms for Volterra equations of the first kind are known. It is generally recognized that Volterra equations of the first kind are much more difficult to solve than Volterra equations of the second kind. Our results confirm this intuition.

We first consider Volterra equations (3) of the second kind. Since an ordinary differential equation is easily convertible to a Volterra integral equation of the second kind, it is only interesting if we require that the function K satisfy the Lipschitz condition. We say K satisfies the *Lipschitz condition* if there exists a constant $L > 0$ such that for all $y, s \in [0, 1]$ and all $t_1, t_2 \in [-1, 1]$

$$|K(y, s, t_1) - K(y, s, t_2)| \leq L \cdot |t_1 - t_2|.$$

If K satisfies the Lipschitz condition, then it is known, through Picard's successive approximation algorithm, that equation (3) has a unique solution φ that is polynomial-space computable. Thus, the complexity of solving equation (3) is precisely the same as the complexity of solving the differential equation (1).

Theorem 2.29 *In the following, (a) $\Rightarrow$ (b) $\Rightarrow$ (c).*

- (a) $P = PSPACE$.
- (b) *If functions K and f of equation (3) are polynomial-time computable and if K satisfies the Lipschitz condition, then the solution φ of equation (3) is polynomial-time computable.*
- (c) $FP = \#P$.

In addition, following from Theorem 2.26, if K only satisfies the Lipschitz condition on a region that polynomially covers the unique solution φ , then we know that $PSPACE$ is also a lower bound for the solution φ .

Next, we consider Volterra equations (3) of the second kind for which the kernel function K only satisfies a *local Lipschitz condition*: for any $y \in [0, 1]$, there exists a constant L_y such that

$$(i) \quad |K(y, s, t_1) - K(y, s, t_2)| \leq L_y \cdot |t_1 - t_2| \text{ for all } s \in [0, 1] \text{ and all } t_1, t_2 \in [-1, 1], \text{ and}$$

$$(ii) \quad \text{there exists a polynomial function } q \text{ such that for all } n \geq 0, \\ 0 \leq y \leq 1 - 2^{-n} \Rightarrow L_y \leq 2^{q(n)}.$$

There are two reasons to consider this type of Volterra equations: first, it demonstrates quantitatively the importance of the Lipschitz condition of the kernel K on the complexity of the solution φ ; and second, this study could be later applied to give a lower bound for Volterra equations of the first kind.

A careful analysis of Picard's successive approximation algorithm for Volterra equations of the second kind (see, for instance, [41]) shows that equation (3) with K satisfying the above local Lipschitz condition is solvable in exponential-space. In the other direction, an exponential-time lower bound has been obtained for such equations through a discrete problem of systems of quadratic equations.

Theorem 2.30 (Ko [30])

(a) *If functions*

$$K : [0, 1]^2 \times [-1, 1] \rightarrow \mathbf{R} \quad \text{and} \quad f : [0, 1] \rightarrow \mathbf{R}$$

are polynomial-time computable and if K satisfies the local Lipschitz condition, then equation (3) has a unique solution φ that is computable in exponential space (i.e., in space 2^{n^k} for some constant $k > 0$).

(b) *There exist polynomial-time computable functions*

$$K : [0, 1]^2 \times [-1, 1] \rightarrow \mathbf{R} \quad \text{and} \quad f : [0, 1] \rightarrow \mathbf{R}$$

such that K satisfies the local Lipschitz condition and the solution φ of equation (3) defined by K and f is unique and has a polynomial modulus, but is complete for EXPTIME.

It is not known whether the gap between the exponential-space upper bound and the exponential-time lower bound can be narrowed.

Now we consider Volterra equations (2) of the first kind. As we pointed out above, no numerical algorithm is known to work for the general case, even if the function K satisfies the Lipschitz condition. Most algorithms attempt to convert a Volterra equation of the first kind into a Volterra equation of the second kind and then solve it by the known algorithms for Volterra equations of the second kind. There are certainly some limitations to this idea. First, more analytical properties on the kernel K , such as differentiability, are required in the conversion. Second, the Lipschitz condition on kernel K is often not preserved by the conversion. Thus the complexity for Volterra

equations of the first kind could be much higher than polynomial-space even if the kernel K satisfies the Lipschitz condition.

A specifically interesting conversion is the following. First we assume that functions K and f are differentiable with respect to variable y . We may then differentiate both sides of equation (2) to obtain

$$K(y, y, \varphi(y)) + \int_0^y \frac{\partial}{\partial y} K(y, s, \varphi(s)) ds = f'(y).$$

Further assume that $K(y, y, \varphi(y)) = K_1(y)\varphi(y)$ with $K_1(y)$ nonvanishing. Then, equation (2) is equivalent to the following equation of the second kind:

$$\varphi(y) = \frac{f'(y)}{K_1(y)} - \int_0^y H(y, s, \varphi(s)) ds, \quad (4)$$

where $H(y, s, t) = \frac{\partial}{\partial y} K(y, s, t)/K_1(y)$. We now may be able to apply, for instance, Picard's successive approximation algorithm to solve this equation. As we pointed out above, however, the new kernel H does not satisfy the Lipschitz condition, even if we assume that the original kernel K does. What we can prove is that if K satisfies the Lipschitz condition then the function H satisfies the local Lipschitz condition. Therefore, by Theorem 2.30, equation (4) has an exponential time lower bound.

Theorem 2.31 (Ko [30])

(a) Assume that functions $K : [0, 1]^2 \times [-1, 1] \rightarrow \mathbf{R}$ and $f : [0, 1] \rightarrow \mathbf{R}$ satisfy the following properties:

- (i) K and f are polynomial-time computable,
- (ii) K satisfies the Lipschitz condition,
- (iii) $K(y, y, t) = K_1(y, y) \cdot t$ and $K_1(y, y) \neq 0$ a.e.,⁷
- (iv) $\frac{\partial}{\partial y} K(y, s, t)$ and $f'(y)$ exist and are polynomial-time computable.

Then, equation (2) has a unique solution φ on $[0, 1]$ that is exponential-space computable.

- (b) There exist functions K and f satisfying conditions (i)–(iv) above, and the unique solution φ of equation (2) has a polynomial modulus but is complete for EXPTIME.

⁷By a.e., we mean that for any $n > 0$, all but finitely many $y < 1 - 2^{-n}$ satisfies $K_1(y, y) \neq 0$.

2.8 Two-dimensional regions

In this section, we study the notion of polynomial-time computability of a two-dimensional bounded, simply-connected region (i.e., a bounded, connected open set with no holes). There are a number of different representations for a two-dimensional region S , each having different applications. Here, we only consider three possible representations.

First, we extend the the notion of recursive approximability to the notion of polynomial-time approximability. Recall from Definition 1.19 that $E_M(n)$ is the set of points in $\mathbf{R}^2$ at which errors might occur for M on input n .

Definition 2.32 A set $S \subseteq \mathbf{R}^2$ is *polynomial-time approximable* if there exists a polynomial-time oracle TM M such that, for each $n \geq 1$, the two-dimensional Lebesgue measure of $E_M(n)$ is bounded by 2^{-n} .

The notion of polynomial-time approximability allows the machine which computes the characteristic function of the set S to have errors but requires that the error probability must be within a predefined bound. Next, we consider the notion of *polynomial-time recognizability* which allows the machine which computes the characteristic function of the set S to have errors only around the boundary of the set S , and hence is useful for regions whose boundaries are Jordan curves. For any set $S \subseteq \mathbf{R}^2$ and any point $\langle x, y \rangle \in \mathbf{R}^2$, we let $\delta(\langle x, y \rangle, S)$ be the distance between $\langle x, y \rangle$ and S , in particular $\delta(\langle x, y \rangle, S) = \min\{|\langle x, y \rangle - \langle u, v \rangle| : \langle u, v \rangle \in S\}$.

Definition 2.33 A set $S \subseteq \mathbf{R}^2$ is *polynomial-time recognizable* if there exists a polynomial-time oracle TM M such that, for any integer $n \geq 1$, $E_M(n) \subseteq \{\langle x, y \rangle : \delta(\langle x, y \rangle, S) \leq 2^{-n}\}$.

The third notion of polynomial-time computability is strictly limited to sets with a polynomial-time computable Jordan curve as a boundary. (We defined the notion of computability of a Jordan curve in Section 1.5. Its extension to polynomial-time computability is straitforward.) We now compare these three notions.

First, is a polynomial-time recognizable set always polynomial-time approximable? The answer is a negative one, and the proof is a simple extension of that of Theorem 1.21.

Theorem 2.34 *There exists a polynomial-time computable Jordan curve Γ such that its interior $\text{Int}(\Gamma)$ is polynomial-time recognizable but not recursively approximable.*

Next, we study whether a polynomial-time approximable set is always polynomial-time recognizable. To answer this question, we need a new concept of probabilistic computation.

In Section 2.2.1, we defined BPP as the class of sets computable in polynomial time by probabilistic TM's. We say two sets A and B are a *BP-pair* if $A \cap B = \emptyset$ and there exists a polynomial-time probabilistic TM M that

- (i) accepts each $x \in A$ with probability $\geq 3/4$, and
- (ii) rejects each $y \in B$ with probability $\geq 3/4$.

Two disjoint sets A and B are called *P-separable* if there exists a set $C \in \mathbf{P}$ such that $A \subseteq C$ and $B \subseteq \overline{C}$. It is not known whether $\text{BPP} = \mathbf{P}$ implies that all BP-pairs are P-separable.

Theorem 2.35 (Chou and Ko [11]) *In the following,*

(a) $\Rightarrow$ (b) $\Rightarrow$ (c) $\Rightarrow$ (d).

- (a) $\text{FP} = \#\mathbf{P}$.
- (b) All BP-pairs are P-separable.
- (c) All polynomial-time approximable subsets of $[0, 1]^2$ are polynomial-time recognizable.
- (d) $\text{BPP} = \mathbf{P}$.

The third result is about the question of whether the interior of a polynomial-time computable Jordan curve is polynomial-time recognizable.

Theorem 2.36 (Chou and Ko [11]) *In the following,*

(a) $\Rightarrow$ (b) $\Rightarrow$ (c).

- (a) $\text{FP} = \#\mathbf{P}$.
- (b) For every polynomial-time computable Jordan curve Γ , its interior $\text{Int}(\Gamma)$ is polynomial-time recognizable.
- (c) $\text{UP} = \mathbf{P}$. i.e., one-way functions do not exist.

Finally we apply the above results to the problem about the measure of the interior of a polynomial-time computable Jordan curve.

Corollary 2.37 *The following are equivalent:*

- (a) $\text{FP} = \#\text{P}$.
- (b) *For every polynomial-time approximable set $S \subseteq [0, 1]^2$, its two-dimensional Lebesgue measure is polynomial-time computable.*
- (c) *For every polynomial-time computable Jordan curve Γ that is rectifiable (i.e., having a finite length), the measure of its interior $\text{Int}(\Gamma)$ is polynomial-time computable.*

Corollary 2.38 *There exists a polynomial-time computable Jordan curve Γ whose interior is polynomial-time recognizable, but its measure is not recursive.*

For other complexity issues about two-dimensional regions, such as the distance between a point and a region and the complexity of computing the winding numbers of a given curve, see [11].

3 Other approaches

As we pointed out in the Introduction, many different approaches have been proposed to study the computational complexity of continuous problems. Each approach seems to aim at some specific domain of problems. We believe that the approach discussed in Section 2 is best for a general complexity theory for real functions, particularly for proving lower bounds based on discrete polynomial-time complexity theory. However, some specific problems and some upper bound results may be better studied in different computational models. In this section, we give a brief introduction to a few other interesting approaches. We limit ourselves to presenting an informal computational model and an example of results in each approach. The interested readers are referred to the technical papers in the bibliography.

3.1 Real random access machines

In computational linear algebra and computational geometry, as well as in mathematical programming, one often uses the real random access machine (real RAM) as the computational model (see [9]). A real RAM is a random access machine in which a real number, with the infinite precision, can be

stored in a single address. The machine is able to perform certain basic operations on real numbers, such as addition, multiplication, division by nonzeros and comparison, where real numbers can be randomly accessed by their symbolic names. The complexity of a real RAM is usually measured by the number of operations performed on a specific input. A typical question studied in this model is the matrix multiplication problem: given two $n \times n$ matrices A and B , compute their product $C = A \times B$. (For recent progress on this problem, see, for instance, [13].)

As we pointed out earlier, this model of computation assumes the availability of the infinite-precision operations which is completely incompatible with the approach in Section 2. A computational theory of real functions based on the real RAM model is closer to *symbolic computation* than to numerical computation. However, in Sections 3.3 and 3.4, we will present two complexity theories of real functions based on this model.

3.2 Representing real functions by programs

Since our interests are focused on polynomial-time computable real functions, a numerical operator which maps such a function to a real number may be considered as a function that maps a program, in a fixed computational model, to a real number. This idea is close to the approach in symbolic computation, except that here we allow hybrid numerical and symbolic computation in a single computational model.

3.2.1 Turing Machine representation

In Corollary 2.4 we have seen that each polynomial-time computable real function f could be represented by a modulus function m and a discrete function φ mapping rationals to rationals. Thus, such a function could be represented by a TM, with the modulus functions predefined. For instance, the integration problem in this approach is formulated in the following form:

Integration Problem. Given a TM M and two integers $i, n > 0$, find a rational number r such that if M computes a real function $f : [0, 1] \rightarrow \mathbf{R}$ with a modulus function $m(k) = k^i$ then $|r - \int_0^1 f| \leq 2^{-n}$.

Note that this is not a purely symbolic computation problem, since we ask for a real number as the output. Also, this problem is stronger than

the polynomial-time computability approach of Section 2: we ask here for a general integration algorithm for all polynomial-time computable functions with a modulus function $m(k) = k^i$, but in Section 2, we allowed different integration algorithms for different functions.

One of the interesting results obtained in this approach is the constructive fixed-point theorem. The classical d -dimensional Brouwer fixed-point theorem states that a continuous function f from Δ_d to Δ_d must have a fixed point $x \in \Delta_d$ in the sense that $f(x) = x$, where Δ_d is the d -dimensional unit cube. Hirsh et al. [19] have shown that in the real RAM model, the functional mapping a real function to one of its fixed points is not polynomial-time computable. As we pointed out in Section 2, such a lower bound result on functionals does not reveal much of its inherent complexity. Papadimitriou [49] considered the fixed-point theorem in the model of representing real functions by TM's.

Fixed-Point Problem. Let d be an integer $d \geq 1$. Given a TM M computing a real function f from Δ_d to Δ_d , with a linear modulus function, and an integer n , find a point $x \in \Delta_d$ such that $|f(x) - x| \leq 2^{-n}$.

Note that the above question asks for an *approximate* fixed point for each input n , and these approximate fixed points are not required to converge to a single fixed point. Thus, this question is weaker than the question of computing the functional of mapping real functions to their fixed points. To characterize the complexity of this problem, Papadimitriou [49] defined a new discrete complexity class PDLF which contains all functions that are defined by a local search strategy similar to that of Sperner's lemma for the fixed-point theorem. We omit here the formal definition of PDLF. It is known that $\text{FP} \subseteq \text{PDLF} \subseteq \text{TFNP}$, where TFNP denotes the class of all total functions that are computable by nondeterministic TM's. Papadimitriou [49] showed some evidence that this class PDLF is probably not equal to FP nor to TFNP.

Theorem 3.1 *The above fixed-point problem is complete for PDLF, when $d \geq 3$.*

It is obvious that the fixed-point problem in the case $d = 1$ is polynomial-time solvable. It is however not known whether the fixed-point problem in the case $d = 2$ is polynomial-time solvable or is complete for PDLF.

It is worth pointing out that this approach is very closely related to the approach presented in Section 2. For instance, the ideas of the proof of the above theorem can also be formulated in the complexity theory of Section 2. In the following, we say a function $f : \Delta_d \rightarrow \Delta_d$ is *well-conditioned* to mean that its approximate fixed points x satisfying $|f(x) - x| \leq \epsilon$ are actually close to some real fixed points y satisfying $f(y) = y$.

Corollary 3.2 (Ko [31]) *Let $d \geq 3$. The following are equivalent:*

- (a) PDLF = FP.
- (b) *For every well-conditioned $f : \Delta_d \rightarrow \Delta_d$ that is computable in polynomial-time and has a unique fixed point x in Δ_d , x is polynomial-time computable.*

A weaker result for the case $d = 2$ is also obtained in [31].

3.2.2 Real-valued circuits

In addition to the Turing machine representation, a real function could also be represented by real-valued circuits. Since real-valued circuits are just an alternative representation to the real RAM model, this approach is more closely related to symbolic computation than to numerical computation. Indeed, the functions that could easily be represented by real-valued circuits are just rational functions f_1/f_2 , where f_1 and f_2 are polynomial functions.

Hoover [21], however, found a very interesting relation between the real-valued circuit representation and the boolean-valued circuit representation for real functions. Namely, he showed that the class of polynomial-time computable real functions, in the sense of Definition 2.2, coincides with the class of functions that are *sup-approximable* by a uniform family of real-valued circuits with polynomial-size and polynomial-magnitude. In the above, a family $\{c_{n,j}\}$ of real-valued circuits is of *polynomial-size* if the number of gates in $c_{n,j}$ is bounded by a polynomial in $n + j$; it is of *polynomial-magnitude* if for any input $x \in [-2^j, 2^j]$, each gate in $c_{n,j}$ outputs a real number in $[-2^{p(n+j)}, 2^{p(n+j)}]$ for some fixed polynomial p ; it is a *uniform family* if there is a polynomial-time TM computing $c_{n,j}$ from (n, j) . Also, a real function f is *sup-approximated* by a family $\{c_{n,j}\}$ of real-valued circuits if for all $x \in [-2^j, 2^j]$, $|f(x) - c_{n,j}(x)| \leq 2^{-n}$.

The above characterization of Hoover gives us a weak form of the polynomial-time version of Wierstrass approximation theorem (see Theorem 1.15 and [29]), and, more importantly, provides a transformation between symbolic computation and numerical computation. For instance, Kaltofen [23] asked whether the integration of polynomial functions is computable in polynomial time, provided that polynomial functions are given by straight-line programs. Using Hoover's characterization, it is immediate that this question is as difficult as the numerical integration problem and hence is $\#P$ -complete.

3.3 Information-based complexity theory

Traub et al. [62] established the information-based complexity theory as a general theory to study approximately solved numerical problems (see also [63] and [68]). The main feature of the theory is to distinguish the notion of *information complexity* from the notion of *computational* (or sometimes called *combinatorial*) *complexity* of a problem. Consider, for instance, the integration problem. In order to compute an approximate value of $\int_0^1 f(t) dt$ with an error bounded by ϵ , we must first evaluate f at some points $x_1, \dots, x_n \in [0, 1]$, and then compute a real number y from the values $f(x_1), \dots, f(x_n)$ such that $|y - \int_0^1 f(t) dt| < \epsilon$. We will call the amount of function evaluations of $f(x_i)$ necessary to obtain the approximation y as the *information complexity* of the integration problem, and call the amount of time required to compute y from the given values $f(x_1), \dots, f(x_n)$ as the computational complexity. Traub et al. [62] demonstrated that information complexity by itself is often a dominating complexity measure for many numerical problems.

Quantitatively, the information complexity of a problem is measured by the *radius of information* required to solve the problem. Informally speaking, for any given problem, we can only collect a finite amount of information $N = N(f)$ about the given input function f . From this set N of information, there are potentially infinitely many functions g in the domain of the input functions that have the same set of information $N(g) = N$. The radius $r(N)$ of information N is defined to be the radius of the minimum *ball* that contains all these functions g , where the measure for the radius depends on the problem. A nice general theorem in this theory states that the information N is strong enough to solve the problem with an error bounded by ϵ with respect to all functions in the domain (disregarding the computational complexity) if and only if the radius $r(N)$ of information N is less than ϵ .

Using this theorem, we can compute the *optimal information* for a numerical problem, and the *optimal algorithm* with respect to the optimal information. Consider again the integration problem. Assume that the domain consists of all functions f such that $|f'(x)| \leq 1$ for all $x \in [0, 1]$. Then, it can be proved that the optimal information for computing $\int_0^1 f(t) dt$ within an error $1/4n$ is taking $f(x_i)$ at equi-distant points $1/2n, 3/2n, \dots, (2n-1)/2n$, and the radius of this set of information is $1/4n$. The worst-case error $1/4n$ could be achieved simply by the Riemann sum of the sample values $f((2i-1)/2n)$, $i = 1, \dots, n$.

Comparing this complexity theory with the theory of polynomial-time analysis of Section 2, we notice the following differences. First, as we pointed out above, information-based complexity emphasizes information complexity and generally disregards computational complexity. In the theory of polynomial-time analysis, both complexity measures are included in the consideration, and therefore the domain of the input functions is often restricted to polynomial-time computable functions. Second, the computational model used in the information-based complexity theory is close to the real RAM model with some new, nonconventional operations. Traub et al. [62] pointed out that it is important to distinguish between different types of information such as the function evaluation $f(x)$, the derivative evaluation $f'(x)$, or sometimes even the integral evaluation $\int_0^1 f$. For different problems, one may assume weak or strong analytic properties of the input functions, and may decide to use inexpensive or costly information to solve the problem. Thus, we may imagine an extended real RAM model with oracles, in which one may assume the availability of different types of information given in the form of oracles, and the cost of collecting different types of information is measured separately (see [48]). Third, the lower bound achieved in the information-based complexity theory is basically by the adversary argument, and in the theory of Section 2, the technique of NP-completeness theory is applied.

Among the similarities between the two theories, we would like to point out that many notions and techniques of discrete algorithm theory could be applied to the information-based complexity theory. Among them, the notions of average-case complexity, parallel computation and probabilistic algorithms have been studied in the theory. Furthermore, if the real RAM model uses the logarithmic cost measure instead of the unit cost measure, then the two models are even closer (see [10] and [67]).

3.4 A theory of NP-completeness for algebraic computation

Blum et al. [7] recently introduced a new theory of NP-completeness based on an algebraic computational model that is very close to the real RAM model. Since the only operations allowed in such a machine are arithmetic operations on reals, this theory is naturally more suitable for algebraic computational problems. There are two novel features of this theory, however, that make it different from the classical algebraic complexity theory. First, the machine is allowed to have some fixed real numbers as its parameters; that is, a finite number of fixed real numbers are allowed to be prestored in the machine. As a consequence, there exist an uncountable number of machines (unlike any other computational models used in discrete or continuous computational theory). Second, nondeterministic machines are defined in such a way that a nondeterministic node of a machine may guess a real number with infinite precision (instead of being able to guess only a single bit in nondeterministic TM's). This means that the nondeterministic node has an uncountable number of computational successors and, in general, cannot be simulated by deterministic machines.

To be more precise, a (deterministic) machine M in this theory is a real RAM with arithmetic operations and the compare-and-branch operation that compares two real numbers and branches to new instructions according to the comparison result. Let $\mathbf{R}^\infty$ denote the set of all infinite sequences of real numbers with only a finite number of nonzeros. A sequence $\vec{x} = (x_1, \dots, x_n, 0, \dots)$ in $\mathbf{R}^\infty$ is of *size* n , and we write $\ell(\vec{x}) = n$, if n is the largest integer such that $x_n \neq 0$. A *decision problem* A is a subset of $\mathbf{R}^\infty$. We say a decision problem A is *polynomial-time computable*, and write $A \in \mathbf{P}_\mathbf{R}$, if there is a machine M that solves an instance $\vec{x}$ of A such that $\vec{x} \in A$ if and only if $M(\vec{x})$ outputs 1 and that $M(\vec{x})$ halts in time $O(\ell(\vec{x})^k)$ for some constant $k > 0$. A decision problem A is in $\mathbf{NP}_\mathbf{R}$ if there exists a polynomial-time machine M such that for each instance $\vec{x} \in \mathbf{R}^\infty$, $\vec{x} \in A$ if and only if there exists a sequence $\vec{y} \in \mathbf{R}^\infty$, $\ell(\vec{y}) \leq O(\ell(\vec{x})^k)$ for some constant $k > 0$, such that $M(\vec{x}, \vec{y})$ outputs 1. A decision problem A is *NP_R-complete* if $A \in \mathbf{NP}_\mathbf{R}$ and for all $B \in \mathbf{NP}_\mathbf{R}$ there exists a polynomial-time machine M such that for each $\vec{x} \in \mathbf{R}^\infty$, $M(\vec{x})$ outputs a $\vec{y} \in \mathbf{R}^\infty$ such that $\vec{x} \in A$ if and only if $\vec{y} \in B$. It can be seen that these definitions are motivated by the discrete NP-completeness theory.

Blum et al. [7] demonstrated a few problems that are $\text{NP}_{\mathbf{R}}$ -complete. Among them, the following problem is the most fundamental one.

Degree-4 Algebraic Variety Problem. Given a degree-4 polynomial φ over n variables $x_1, \dots, x_n$, determine whether there exists an $\vec{x} \in \mathbf{R}^\infty$, $\ell(\vec{x}) \leq n$, such that $\varphi(\vec{x}) = 0$.

Theorem 3.3 *The degree-4 algebraic variety problem is $\text{NP}_{\mathbf{R}}$ -complete.*

From these $\text{NP}_{\mathbf{R}}$ -complete problems, it is established that whether $\mathbf{P}_{\mathbf{R}}$ is equal to $\text{NP}_{\mathbf{R}}$ is the fundamental question in this theory. Although this question is based on a theory using the real RAM model, we may treat the theory as a theory of symbolic computation and hence as a subtheory of the classical discrete complexity theory. Since it is not known whether $\mathbf{P}_{\mathbf{R}} = \text{NP}_{\mathbf{R}}$, it would be interesting to know whether it is equivalent to some well-known open questions in discrete complexity theory. At present, this question is not known to be equivalent to the question $\mathbf{P} = \mathbf{NP}$ in the classical NP -completeness theory. The best known relation between this question and questions in discrete complexity theory is that if $\mathbf{P} = \text{PSPACE}$ then $\mathbf{P}_{\mathbf{R}} = \text{NP}_{\mathbf{R}}$. For further development of this theory, see the forthcoming book [6].

References

- [1] O. Aberth, Analysis in the computable number field, J. Assoc. Comput. Mach., **15** (1968) 275–299.
- [2] C. T. H. Baker, The Numerical Treatment of Integral Equations, (Oxford Univ. Press, 1977).
- [3] J. L. Balcázar, J. Díaz and J. Gabarró, Structural Complexity, Vol. 1, 2, (Springer-Verlag, Berlin, 1988, 1990), (2nd. rev. edn. 1995).
- [4] E. Bishop, Foundations of Constructive Analysis, (McGraw-Hill, New York, Toronto, London, 1967).
- [5] E. Bishop and D. S. Bridges, Constructive Analysis, Grundlehren Math. Wiss., **279** (1985).

- [6] L. Blum, F. Cucker, M. Shub and S. Smale, *Complexity and Real Computation*, (Springer-Verlag, NY, 1997).
- [7] L. Blum, M. Shub and S. Smale, On a theory of computation and complexity over the real numbers: NP-completeness, recursive functions and universal machines, *Bull. Amer. Math. Soc. (N.S.)*, **21** (1989) 1–46.
- [8] R. V. Book, Tally languages and complexity classes, *Inform. Contr.*, **26** (1974) 186–193.
- [9] A. Borodin and I. Munro, *The computational complexity of algebraic and numeric problems*, (Elsevier, New York, 1975).
- [10] V. Brattka and P. Hertling, Feasible real random access machines, in: *Theory and Practice of Informatics, SOFSEM '96*, K. G. Jeffrey, J. Král and M. Bartošek, (eds.), *Lecture Notes in Comput. Sci.*, **1175** (1996) 335–342.
- [11] A. W. Chou and K. Ko, Computational complexity of two-dimensional regions, *SIAM J. Comput.*, **24** (1995) 923–947.
- [12] S. A. Cook, A taxonomy of problems with fast parallel algorithms, *Inform. Contr.*, **64** (1985) 2–22.
- [13] D. Coppersmith and S. Winograd, Matrix multiplication via arithmetic progressions, *J. Symbolic Comput.*, **9** (1990) 251–280.
- [14] K. J. Falconer, *Fractal Geometry, Mathematical Foundations and Applications*, (John Wiley and Sons, NY, 1991).
- [15] H. M. Friedman, The computational complexity of maximization and integration, *Adv. Math.*, **53** (1984) 80–98.
- [16] M. R. Garey and D. S. Johnson, *Computers and Intractability, a guide to the theory of NP-completeness*, (W. H. Freeman, San Francisco, 1979).
- [17] A. Grzegorzczuk, On the definitions of computable real continuous functions, *Fund. Math.*, **44** (1957) 61–71.
- [18] A. Grzegorzczuk, Some approaches to constructive analysis, in: *Constructivity in Mathematics*, (Proc. Colloq., Amsterdam, Aug. 26–31, 1957), A. Heyting, (ed.), *Stud. Logic Found. Math.*, (1959) 43–61.

- [19] M. D. Hirsh, C. H. Papadimitriou and S. A. Vavasis, Exponential lower bounds for finding Brouwer fixpoints, *J. Complexity*, **5** (1989) 379–416.
- [20] H. J. Hoover, Feasible Constructive Analysis, Ph.D. Thesis, Dept. Comput. Sci., Univ. Toronto, Toronto, Canada, (1987).
- [21] H. J. Hoover, Feasible real functions and arithmetic circuits, *SIAM J. Comput.*, **19** (1990) 182–204.
- [22] J. E. Hopcroft and J. D. Ullman, Introduction to Automata Theory, Languages, and Computation, (Addison-Wesley, Reading, Mass., 1979).
- [23] E. Kaltofen, Single-factor Hensel lifting and its application to the straight-line complexity of certain polynomials, in: Proc. 19th. ACM Sympos. Theory of Computing, (NYC, May 25–27, 1987), M. Sipser, (ed.), (ACM Press, NY, NY, 1987), 443–452.
- [24] K. Ko, The maximum value problem and NP real numbers, *J. Comput. System Sci.*, **24** (1982) 15–35.
- [25] K. Ko, On the computational complexity of ordinary differential equations, *Inform. Contr.*, **58** (1983) 157–194.
- [26] K. Ko, On the definitions of some complexity classes of real numbers, *Math. Systems Theory*, **16** (1983) 95–109.
- [27] K. Ko, Approximation to measurable functions and its relation to probabilistic computation, *Ann. Pure Appl. Logic*, **30** (1986) 173–200.
- [28] K. Ko, Inverting a one-to-one real function is inherently sequential, in: Feasible Mathematics, (Proc. Workshop, June 16–18, 1989, Cornell Univ. Ithaca, NY), S. Buss and P. J. Scott (eds.), *Progr. Comput. Sci. Appl. Logic*, **9** (1990) 239–257.
- [29] K. Ko, Complexity theory of real functions, *Progr. Theoret. Comput. Sci.*, (1991).
- [30] K. Ko, On the computational complexity of integral equations, *Ann. Pure Appl. Logic*, **58** (1992) 201–228.
- [31] K. Ko, Computational complexity of fixed points and intersection points, *J. Complexity*, **11** (1995) 265–292.

- [32] K. Ko, A polynomial-time computable curve whose interior has a nonrecursive measure, *Theoret. Comput. Sci.*, **145** (1995) 241–270.
- [33] K. Ko, On the computability of fractal dimensions and julia sets, *Ann. Pure Appl. Logic*, (to appear).
- [34] K. Ko and H. M. Friedman, Computational complexity of real functions, *Theoret. Comput. Sci.*, **20** (1982) 323–352.
- [35] K. Ko and K. Weihrauch, On the measure of two-dimensional regions with polynomial-time computable boundaries, in: *Computational Complexity*, (Proc. 11th. Annual IEEE Conf. Computational Complexity), S. Homer and J.-Y. Cai, (eds.), (IEEE Comput. Soc. Press, Los Alamitos, CA, 1996), 150–159.
- [36] D. Lacombe, Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles I, *C. R. Acad. Sci. Paris*, **240** (1955) 2478–2480.
- [37] D. Lacombe, Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles II, *C. R. Acad. Sci. Paris*, **241** (1955), 13–14.
- [38] D. Lacombe, Extension de la notion de fonction récursive aux fonctions d'une ou plusieurs variables réelles III, *C. R. Acad. Sci. Paris*, **241** (1955) 151–153.
- [39] G. Metakides and A. Nerode, The introduction of nonrecursive methods into mathematics, in: *The L. E. J. Brouwer Centenary Sympos.*, (Noordwijkerhout, June 8–13, 1981), A. S. Troelstra and D. van Dalen, (eds.), *Stud. Logic Found. Math.*, **110** (1982) 319–335.
- [40] G. Metakides, A. Nerode and R. A. Shore, Recursive limits on the Hahn-Banach Theorem, in: *Errett Bishop: reflections on him and his research*, (San Diego, Calif., 1983), M. Rosenblatt, (ed.), *Contemp. Math.*, **39** (1985) 85–91.
- [41] R. K. Miller, *Nonlinear Volterra Integral Equations*, Mathematics Lecture Note Series, (W. A. Benjamin, Menlo Park, California, 1971).
- [42] W. Miller, Recursive function theory and numerical analysis, *J. Comput. System Sci.*, **4** (1970) 465–472.

- [43] A. Mostowski, On computable sequences, *Fund. Math.*, **44** (1957) 37–51.
- [44] A. Mostowski, On various degrees of constructivism, in: *Constructivity in Mathematics*, (Proc. Colloq., Amsterdam, Aug. 26–31, 1957), A. Heyting, (ed.), *Stud. Logic Found. Math.*, (1959) 178–194.
- [45] J. Myhill, A recursive function, defined on a compact interval and having a continuous derivative that is not recursive, *Michigan Math. J.*, **18** (1971) 97–98.
- [46] C. A. Neff, Specified precision polynomial root isolation is in NC, in: *Foundations of Computer Science*, (Proc. 31st. Annual Sympos. Foundations Comput. Sci., St. Louis, MO, Oct. 22–24, 1990), FOCS-90, (IEEE Comput. Soc. Press, Los Alamitos, CA, 1990), 152–162.
- [47] A. Nerode and W. Q. Huang, An application of pure recursion theory to recursive analysis (Chinese), *Acta Math. Sinica*, **28** (1985) 625–636.
- [48] E. Novak, The real number model in numerical analysis, *J. Complexity*, **11** (1995) 57–73.
- [49] C. H. Papadimitriou, On graph-theoretic lemmata and complexity classes, in: *Foundations of Computer Science*, (Proc. 31st. Annual Sympos. Foundations Comput. Sci., St. Louis, MO, Oct. 22–24, 1990), FOCS-90, (IEEE Comput. Soc. Press, Los Alamitos, CA, 1990) 794–801.
- [50] C. H. Papadimitriou and J. Tsitsiklis, Intractable problems in control theory, *SIAM J. Contr. Optim.*, **24** (1986) 639–654.
- [51] M. B. Pour-El and J. Caldwell, On a simple definition of computable functions of a real variable — with applications to functions of a complex variable, *Z. Math. Logik Grundlag. Math.*, **21** (1975) 1–19.
- [52] M. B. Pour-El and J. I. Richards, Differentiability properties of computable functions — a summary, *Acta Cybernet.*, **4** (1978/79) 123–125.
- [53] M. B. Pour-El and J. I. Richards, A computable ordinary differential equation which possesses no computable solution, *Ann. Math. Logic*, **17** (1979) 61–90.
- [54] M. B. Pour-El and J. I. Richards, Computability and noncomputability in classical analysis, *Trans. Amer. Math. Soc.*, **275** (1983) 539–560.

- [55] M. B. Pour-El and J. I. Richards, Noncomputability in analysis and physics: a complete determination of the class of noncomputable linear operators, *Adv. Math.*, **48** (1983), 44–74.
- [56] M. B. Pour-El and J. I. Richards, Computability in Analysis and Physics, *Perspect. in Math. Logic*, (1989).
- [57] R. M. Robinson, Review of “Rekursive Funktionen” by R. Péter, *J. Symbolic Logic*, **16** (1951) 280–282.
- [58] N. A. Šanin, Constructive real numbers and constructive functional spaces (Russian), *Trudy Mat. Inst. Steklov.*, **67** (1962) 15–294; [translated: by E. Mendelson, *Transl. Math. Monographs*, **21** (1968)].
- [59] W. Savitch, Relationships between nondeterministic and deterministic tape complexities, *J. Comput. System Sci.*, **4** (1970) 177–192.
- [60] A. Schönhage, The fundamental theorem of algebra in terms of computational complexity, *Mathematisches Institut der Universität Tübingen*, West Germany, (preprint).
- [61] E. Specker, Der Satz vom Maximum in der rekursiven Analysis, in: *Constructivity in Mathematics*, (Proc. Colloq., Amsterdam, Aug. 26–31, 1957), A. Heyting, (ed.), *Stud. Logic Found. Math.*, (1959) 254–265.
- [62] J. Traub, G. Wasilkowski and H. Woźniakowski, Information-based complexity, *Computer Science and Scientific Computing*, (Academic Press, Inc., Boston, MA, 1988).
- [63] J. Traub and H. Woźniakowski, Recent progress in information-based complexity, *Bull. European Assoc. Theoret. Comput. Sci.*, **51** (1993) 141–154.
- [64] F. G. Tricomi, *Integral Equations*, (Interscience, New York, 1957).
- [65] K. Weihrauch, A simple introduction to computable analysis, *Informatik Berichte 171*, FernUniversität Hagen, Hagen, July 1995, 2nd. edn..
- [66] K. Weihrauch, A foundation for computable analysis, in: *Combinatorics, Complexity, and Logic* (Singapore), (Proc. 1st. Internatl. Conf. Discrete Math. Theoret. Comput. Sci., DMTCS '96, Auckland, New Zealand, Dec. 1996), D. S. Bridges, C. S. Calude, J. Gibbons, S. Reeves and I. H. Witten, (eds.), *Discrete Math. Theoret. Comput. Sci.*, (1997) 66–89.

- [67] K. Weihrauch, A refined model of computation for continuous problems, *J. Complexity*, **14** (1998) 102–121.
- [68] A. G. Werschulz, *The computational complexity of differential and integral equations: an information-based approach*, Oxford Mathematical Monographs, (Oxford Univ. Press, NY, 1991).

Ker-I Ko
Department of Computer Science
State University of New York at Stony Brook
Stony Brook, NY 11794
keriko@cs.sunysb.edu

This Page Intentionally Left Blank

Chapter 20

Generally Constructive Boolean Algebras

S. P. Odintsov*

Contents

- Introduction
- 1. Preliminaries
- 2. Trees and Boolean algebras
- 3. Arithmetical hierarchy of Boolean algebras
- 4. Restricted theories of constructive Boolean algebras
- 5. Constructive Boolean algebras with decidability conditions on ultrafilters
- 6. Constructivity beyond first order logic
- References

Introduction

Algorithmic aspects of countable Boolean algebras have been studied by many mathematicians very intensively, and their efforts were very fruitful. A systematic account of results on constructive Boolean algebras is contained in the book by Goncharov [17] and in the survey paper by Remmel [33]. In the following article, we will focus on results which lie outside the main field of research. We reflect several attempts to generalize the fundamental notions of constructive model theory, as well as results on Boolean algebras that are obtained in this context. Countable Boolean algebras are traditionally considered a convenient playground for constructive model theory, and we consider the results of this survey as an example for this assertion.

*This work is supported by RFFR, grant No. 96-01-01552.

In the following, we illustrate how one can generalize the basic notions of constructive model theory.

Let $\mathfrak{M} \triangleq \langle |\mathfrak{M}|, P_0, \dots, P_n, f_0, \dots, f_k, c_0, \dots, c_m \rangle$ be a countable model of finite signature. A numeration [10] of $\mathfrak{M}$ is a map ν from the set of natural numbers ω onto the universe $|\mathfrak{M}|$ of $\mathfrak{M}$ such that the basic operations admit an effective presentation, i.e., for any $i \leq k$, there exists a general recursive function g_i such that

$$\nu g_i(x_0, \dots, x_{m_i}) = f_i(\nu x_0, \dots, \nu x_{m_i}),$$

where m_i is the arity of f_i . The pair $(\mathfrak{M}, \nu)$ is called a numerated model. A numeration ν is called a constructivization of $\mathfrak{M}$ if the numeration equivalence $\sim_\nu \triangleq \{(x, y) \mid \nu x = \nu y\}$, and the sets

$$P_i^\nu \triangleq \{(x_0, \dots, x_{k_i}) \mid \mathfrak{M} \models P(\nu x_0, \dots, \nu x_{k_i})\},$$

$i \leq n$, are recursive. The pair $(\mathfrak{M}, \nu)$ is called a constructive model in this case. The class of recursive sets is very small, but we can weaken the condition on the sets $\sim_\nu, P_i^\nu, i \leq n$, "to be recursive".

If sets $\sim_\nu, P_i^\nu, i \leq n$, are in $\mathcal{A}$, and $\mathcal{A}$ is the family of sets natural from the point of view of algorithm theory, then ν and $(\mathfrak{M}, \nu)$ are called an $\mathcal{A}$ -constructivization and an $\mathcal{A}$ -constructive model, respectively. We can substitute for $\mathcal{A}$ the classes of arithmetic or analytic hierarchy, Turing and m -degrees, etc.. So we may talk about arithmetic and hyperarithmetic models, for example.

A numerated model $(\mathfrak{M}, \nu)$ is called strongly constructive if the set

$$\{(s, \ell_1, \dots, \ell_k) \mid s \text{ is the Gödel number of a first order formula } \varphi(x_1, \dots, x_k) \text{ with } k \text{ free variables, and } \mathfrak{M} \models \varphi(\nu \ell_1, \dots, \nu \ell_k)\}$$

is recursive.

Changing the class of formulae in this definition, we obtain another natural generalization of the notion of a constructive model. We may speak about $\mathcal{S}$ -constructive models, where $\mathcal{S}$ is either a proper subclass of a first order language, or a fragment of second order logic. The (strong) constructivity of a model refers to the existence of an algorithm that verifies the truth of quantifier-free (first order) formulae on this model. It is useful to investigate the existence of such algorithms for other classes of formulae. We select the class of formulae which reflects the algorithmic properties of the model most adequately.

The main part of this paper is devoted to the following topics:

First, the examination of arithmetic Boolean algebras, i.e., Boolean algebras which admit a numeration with arithmetic numeration equivalence. In particular, the structure of the arithmetical hierarchy of numerated Boolean algebras will be described.

Second, the study of restricted theories of constructive Boolean algebras, i.e., theories which include only formulae with a restricted number of alternations of quantifiers.

Third, the constructions of Boolean algebras with various decidability conditions on their ultrafilters, and the connection between these results and the general theory of constructive models.

Finally, we mention results about Δ_1^1 -constructive Boolean algebras and about constructivizability of Boolean algebras in some extensions of first order language.

1 Preliminaries

We use the standard terminology of recursion theory [35] and model theory [2].

Let σ be a finite signature, $\mathcal{F}$ a decidable fragment of a second order language,

$$\{\varphi_n(v_0, \dots, v_{k_n-1})\}_{n < \omega}$$

a Gödel numbering of all formulae of the fragment $\mathcal{F}$ where $v_0, \dots, v_{k_n-1}$ is the list of all free variables of φ_n , and let $\langle \cdot, \dots, \cdot \rangle$ be a fixed coding function for finite sequences of natural numbers.

We consider now a numerated model $(\mathfrak{M}, \nu)$ (see the introduction) of signature σ . Let

$$\mathcal{D}_\nu(\mathfrak{M}, \mathcal{F}) \triangleq \{ \langle n, x_0, \dots, x_{k_n-1} \rangle \mid \mathfrak{M} \models \varphi_n(\nu x_0, \dots, \nu x_{k_n-1}) \},$$

and $\mathcal{A}$ be a class of subsets of ω closed with respect to recursive isomorphisms.

Definition 1.1 A model $\mathfrak{M}$ is called $\mathcal{A}$ - $\mathcal{F}$ -constructivizable if there exists a numeration ν of $\mathfrak{M}$ such that $\mathcal{D}_\nu(\mathfrak{M}, \mathcal{F}) \in \mathcal{A}$. In this case, the numerated model $(\mathfrak{M}, \nu)$ is called $\mathcal{A}$ - $\mathcal{F}$ -constructive, and ν is an $\mathcal{A}$ - $\mathcal{F}$ -constructivization of $\mathfrak{M}$.

If $\mathcal{F}$ is the set of quantifier-free formulae, or the set of formulae with $\leq n$ alternations of quantifiers, or the set of all first order formulae, then instead of “ $\mathcal{A}$ - $\mathcal{F}$ -constructive” (or “ $\mathcal{A}$ - $\mathcal{F}$ -constructivizable”), we use the words “ $\mathcal{A}$ -constructive” (or “ $\mathcal{A}$ -constructivizable”), “ $\mathcal{A}$ - n -constructive” (or “ $\mathcal{A}$ - n -constructivizable”), and “ $\mathcal{A}$ -strongly constructive” (or “ $\mathcal{A}$ -strongly constructivizable”), respectively. If $\mathcal{A}$ is the class of all recursive sets, we omit the prefix $\mathcal{A}$. This corresponds to the usual terminology of constructive model theory. If A is a set, we say “ A -constructive” (or “ A -constructivizable”) instead of “ $\{C \mid C \leq_T A\}$ -constructive” (or “ $\{C \mid C \leq_T A\}$ -constructivizable”), where $\leq_T$ denotes Turing reducibility.

The Ershov school only used numerations. Therefore we give an alternative definition.

Let $\mathfrak{M}$ be a model of signature σ with the universe ω , i.e., $|\mathfrak{M}| = \omega$. We put

$$\mathcal{D}(\mathcal{F}, \mathfrak{M}) \Leftarrow \{ \langle n, x_0, \dots, x_{k_n-1} \rangle \mid \mathfrak{M} \models \varphi_n(\nu x_0, \dots, \nu x_{k_n-1}) \}.$$

Definition 1.2 A model $\mathfrak{M}$ is called $\mathcal{A}$ - $\mathcal{F}$ -constructive if $\mathcal{D}(\mathcal{F}, \mathfrak{M}) \in \mathcal{A}$.

If $\mathfrak{M} \simeq \mathfrak{N}$, $|\mathfrak{N}| = \omega$, and $\mathfrak{N}$ is $\mathcal{A}$ - $\mathcal{F}$ -constructive, then $\mathfrak{N}$ is called an $\mathcal{A}$ - $\mathcal{F}$ -presentation of $\mathfrak{M}$.

These definitions are in some sense equivalent. For instance, we have the following.

Lemma 1.1 Let $\mathfrak{M}$ be a model of functional signature. $\mathfrak{M}$ is $\mathcal{A}$ -constructivizable if and only if $\mathfrak{M}$ has an $\mathcal{A}$ -presentation.

$\mathfrak{M}$ is a model of functional signature, i.e., an algebra. Any model of functional signature is isomorphic to a factor-algebra of the free algebra of this signature, which has an effective presentation in terms of basic operations. The lemma is a simple consequence of this fact.

Let $\mathfrak{M}$ be a countable system of signature σ . The $\mathcal{F}$ -spectrum of $\mathfrak{M}$ is a family $\text{Spec}(\mathcal{F}, \mathfrak{M})$ of Turing degrees such that $a \in \text{Spec}(\mathcal{F}, \mathfrak{M})$ if and only if there exists an a - $\mathcal{F}$ -presentation $\mathfrak{N}$ of the system $\mathfrak{M}$.

Definition 1.3 The degree of $\mathcal{F}$ -constructivizability of $\mathfrak{M}$ is the least element $\deg(\mathcal{F}, \mathfrak{M})$ of the family $\text{Spec}(\mathcal{F}, \mathfrak{M})$.

This definition was suggested by L. Richter [34]. Richter also established that not every Boolean algebra has a degree of constructivizability. In order to prove this, one uses the following idea. For any Boolean algebra $\mathfrak{B}$, one can construct two presentations $\mathfrak{A}_1$ and $\mathfrak{A}_2$ such that $\{\deg(\mathcal{D}(\mathfrak{A}_1)), \deg(\mathcal{D}(\mathfrak{A}_2))\}$ is a minimal pair. Nevertheless, for any degree a , one can construct a system with degree of (strong) constructivizability a (see [37]). Selivanov has suggested the weakening of the Definition 1.3 in order to define the notion of complexity for all systems.

Definition 1.4 The *complexity of $\mathcal{F}$ -constructivizability* of a system $\mathfrak{M}$ is the least ordinal α such that there exists a presentation $\mathfrak{N}$ of $\mathfrak{M}$ with $\mathcal{D}(\mathcal{F}, \mathfrak{M}) \leq_T \mathbf{O}^{(\alpha)}$. Here, $\mathbf{O}^{(\alpha)}$ denotes the α -iteration of the T -jump of the degree of the recursive sets.

It is well known [1, p. 644] that we can consider the sequence of degrees $\{\mathbf{O}^{(\alpha)}\}$ as a sequence which is T -confinal in the class of all Gödel-constructive subsets of ω .

Therefore, any model $\mathfrak{M}$ that has a presentation $\mathfrak{N}$ where $\mathcal{D}(\mathcal{F}, \mathfrak{N})$ is Gödel-constructive has an $\mathcal{F}$ -complexity.

It is clear that a model $\mathfrak{M}$ is Δ_n - $\mathcal{F}$ -constructivizable if and only if the $\mathcal{F}$ -complexity of $\mathfrak{M}$ is $\leq n$. It is also known [37] that the complexities of strong constructivizability of the lattice of r.e. sets and of the ordering of m -degrees are equal to ω .

As usual, we consider Boolean algebras in the signature $\langle \cup, \cap, c, 0, 1 \rangle$. Let $\mathfrak{B} = \langle |\mathfrak{B}|, \cup_{\mathfrak{B}}, \cap_{\mathfrak{B}}, c_{\mathfrak{B}}, 0_{\mathfrak{B}}, 1_{\mathfrak{B}} \rangle$ be a Boolean Algebra (BA), then $\leq_{\mathfrak{B}}$ is the natural ordering of $\mathfrak{B}$ ($x \leq_{\mathfrak{B}} y \Leftrightarrow x \cup_{\mathfrak{B}} y = y$). We omit subscripts whenever it would not lead to confusion.

If A is a subset of $|\mathfrak{B}|$, then $\text{gr}(A)$ is a subalgebra of $\mathfrak{B}$ generated by the elements of A .

Let A be a set, and let $P(A)$ denote its power set. $\mathcal{P}(A)$ is the BA of all subsets of A with the natural operations, and $\mathcal{P}^*(A)$ is the factor algebra of $\mathcal{P}(A)$ with respect to equivalence $\sim^*$, where $B_0 \sim^* B_1$ if and only if $(B_0 \setminus B_1) \cup (B_1 \setminus B_0)$ is a finite set. If $B \in P(A)$, then B^* is the corresponding equivalence class in $\mathcal{P}^*(A) = |\mathcal{P}^*(A)|$.

The product $\prod_{i \in \omega} \mathfrak{B}_i$ of a family of BA's $\{\mathfrak{B}_i \mid i \in \omega\}$ is defined in the usual way. The sum $\sum_{i \in \omega} \mathfrak{B}_i$ is a subalgebra of $\prod_{i \in \omega} \mathfrak{B}_i$ such that $a \in \sum_{i \in \omega} \mathfrak{B}_i$ if and only if almost all components of A are equal to the unit, or almost all components of A are equal to zero.

We use the following standard notations:

$A(\mathfrak{B})$ is the set of atoms of $\mathfrak{B}$, i.e., the set of all minimal non-zero elements of the ordering $\leq_{\mathfrak{B}}$.

$Atl(\mathfrak{B})$ is the set of atomless elements, i.e., elements of $\mathfrak{B}$ that do not contain atoms.

$At(\mathfrak{B})$ is the set of atomic elements, i.e., elements of $\mathfrak{B}$ that do not contain atomless elements.

$F(\mathfrak{B})$ is the Fréchet ideal of the BA $\mathfrak{B}$, i.e., the ideal generated by atoms. An iterated sequence of Fréchet ideals $\{F_{\alpha}(\mathfrak{B}) \mid \alpha \in \text{Ord}\}$ is defined in the following way.

$$F_0(\mathfrak{B}) = \{0_{\mathfrak{B}}\}, \quad F_1(\mathfrak{B}) = F(\mathfrak{B}),$$

$$F_{\alpha+1}(\mathfrak{B}) = \{x \in |\mathfrak{B}| \mid x/F_{\alpha}(\mathfrak{B}) \in F(\mathfrak{B}/F_{\alpha}(\mathfrak{B}))\},$$

$$F_{\beta}(\mathfrak{B}) = \bigcup_{\alpha < \beta} F_{\alpha}(\mathfrak{B}),$$

where β is a limit ordinal.

A BA $\mathfrak{B}$ is said to be atomic (atomless) if $1_{\mathfrak{B}} \in At(\mathfrak{B})$ ($1_{\mathfrak{B}} \in Atl(\mathfrak{B})$). A BA $\mathfrak{B}$ is α -atomic if $\mathfrak{B}/F_{\beta}(\mathfrak{B})$ is atomic for all $\beta < \alpha$.

Let $a \in |\mathfrak{B}|$. Then $[a]_{\mathfrak{B}}$ denotes the principal ideal generated by the element a . Sometimes we shall consider $[a]_{\mathfrak{B}}$ as a BA where the element a is a unit of this BA.

A BA $\mathfrak{B}$ is called superatomic if any subalgebra of $\mathfrak{B}$ is atomic. For any countable superatomic BA $\mathfrak{B}$ [17], there exists a countable ordinal α such that $F_{\alpha+1}(\mathfrak{B}) = |\mathfrak{B}|$, yet $F_{\alpha}(\mathfrak{B}) \neq |\mathfrak{B}|$. α is called the ordinal type of the BA $\mathfrak{B}$, and the pair (α, n) , where n is the number of atoms of $\mathfrak{B}/F_{\alpha}(\mathfrak{B})$, is called the type of the superatomic BA $\mathfrak{B}$.

Let $\mathcal{I}$ be a maximal ideal of a superatomic BA $\mathfrak{B}$. The type of $\mathcal{I}$ is the ordinal α such that $F_{\alpha}(\mathfrak{B}) \subset \mathcal{I}$, yet $F_{\alpha+1}(\mathfrak{B}) \not\subset \mathcal{I}$. The sequence $\{\mathcal{F}_{\alpha}\}$ of Fréchet filters and the type of a maximal filter $\mathcal{F}$ of $\mathfrak{B}$ are defined in a similar way.

Now we define another important sequence of ideals. We will need these ideals to classify Boolean algebras up to elementary equivalence [7, 39]. The

Ershov-Tarski ideal $I(\mathfrak{B})$ of the BA $\mathfrak{B}$ is the ideal generated by the set $\text{At}(\mathfrak{B}) \cup \text{Atl}(\mathfrak{B})$, i.e., each element of $I(\mathfrak{B})$ is a join of atomic and atomless elements. Let

$$I_0(\mathfrak{B}) \rightleftharpoons \{0_{\mathfrak{B}}\}, \quad I_{n+1}(\mathfrak{B}) \rightleftharpoons \{x \in \mathfrak{B} \mid x/I_n(\mathfrak{B}) \in I(\mathfrak{B}/I_n(\mathfrak{B}))\},$$

$$A_{n+1}(\mathfrak{B}) \rightleftharpoons \{x \in \mathfrak{B} \mid x/I_n(\mathfrak{B}) \in A(\mathfrak{B}/I_n(\mathfrak{B}))\},$$

$$\text{Atl}_{n+1}(\mathfrak{B}) \rightleftharpoons \{x \in \mathfrak{B} \mid x/I_n(\mathfrak{B}) \in \text{Atl}(\mathfrak{B}/I_n(\mathfrak{B}))\},$$

$$\text{At}_{n+1}(\mathfrak{B}) \rightleftharpoons \{x \in \mathfrak{B} \mid x/I_n(\mathfrak{B}) \in \text{At}(\mathfrak{B}/I_n(\mathfrak{B}))\}.$$

Proposition 1.2 *The predicates I_n , A_{n+1} , Atl_{n+1} and At_{n+1} , $n \in \omega$, are defined by first order formulae, and have Σ_{4n-} , Π_{4n+1-} , Π_{4n+2-} and Π_{4n+3-} definitions, respectively.*

The elementary characteristic $\text{ch}(a)$ of an element $a \in |\mathfrak{B}|$ is the triple $(\text{ch}_1(a), \text{ch}_2(a), \text{ch}_3(a))$, where

$$\begin{aligned} \text{ch}_1(a) &\rightleftharpoons \begin{cases} \infty, & \text{if } a \notin \bigcup_{n \geq 0} I_n(\mathfrak{B}), \\ n & \text{if } a \in I_{n+1}(\mathfrak{B}) \setminus I_n(\mathfrak{B}) \end{cases} \\ \text{ch}_2(a) &\rightleftharpoons \begin{cases} 0, & \text{if } \text{ch}_1(a) = \infty, \\ n, & \text{if } a/I_m(\mathfrak{B}) \text{ contains } n \text{ distinct atoms,} \\ & \text{and } m = \text{ch}_1(a) < \infty, \\ \infty, & \text{if } a/I_m(\mathfrak{B}) \text{ contains infinitely many distinct atoms,} \\ & \text{and } m = \text{ch}_1(a) < \infty \end{cases} \\ \text{ch}_3(a) &\rightleftharpoons \begin{cases} 0, & \text{if } \text{ch}_1(a) = \infty, \text{ or } \text{ch}_1(a) = m < \infty \text{ and } a/I_m(\mathfrak{B}) \\ & \text{does not contain non-zero atomless elements,} \\ 1, & \text{otherwise} \end{cases} \end{aligned}$$

The elementary characteristic of a BA $\mathfrak{B}$ is the elementary characteristic of its unit element, $\text{ch}(\mathfrak{B}) \rightleftharpoons \text{ch}(1_{\mathfrak{B}})$.

Theorem 1.3 *Two BA's $\mathfrak{A}$ and $\mathfrak{B}$ are elementary equivalent if and only if $\text{ch}(\mathfrak{A}) = \text{ch}(\mathfrak{B})$.*

Corollary 1.4 *The elementary theory $\text{Th}(\mathfrak{B})$ of any BA $\mathfrak{B}$ is decidable.*

Let σ be a signature of the BA's, then $\sigma^* \Leftrightarrow \sigma \cup \{A_n, \text{Atl}_n, \text{At}_n, I_n \mid n \in \omega\}$ is the enrichment of σ by a countable set of unary predicates, which will be interpreted in accordance with the definitions given above.

Theorem 1.5 *The theory of BA's is model-complete in the signature σ^* .*

We conclude this section by mentioning a very useful fact from constructive model theory [11].

Theorem 1.6 *Let T be a model-complete theory. Then each constructive model of T is strongly constructive.*

2 Trees and Boolean algebras

Here we describe the methods of presentations of BA's that we will use later on. Binary trees are very important for this purpose [15, 27, 29].

Let $2^{<\omega}$ be the set of all finite sequences of zeros and ones. The elements of $2^{<\omega}$ are denoted by $\sigma, \tau, \rho, \dots$, and will be called strings. Let $\sigma * \tau$ denote the concatenation of the strings σ and τ , $\sigma \subseteq \tau$ means that $\sigma * \rho = \tau$ for some ρ , $\sigma \mid \tau$ denotes that σ and τ are incomparable with respect to $\subseteq$, and $\text{lh}(\sigma)$ is the length of σ .

It is sometimes convenient to regard a string σ as a map from the set $\{i \mid i < \text{lh}(\sigma)\}$ to $\{0, 1\}$.

Given a partial order T , we say that T is a *tree* if

- (i) T is a lower semilattice,
- (ii) T has a least element,
- (iii) each element of T either has exactly two successors, or is itself maximal, and
- (iv) each initial segment of T is a finite linear order.

The tree T is called *full* if T does not have maximal elements. It is clear that the relation $\subseteq$ defines the structure of a full tree on $2^{<\omega}$.

A subset $P \subseteq 2^{<\omega}$ is called a *segment* if the following condition holds.

$$(\tau \in P \ \& \ \sigma \subseteq \tau) \implies \sigma \in P$$

A subtree of the full tree $2^{<\omega}$ is a segment that is itself a tree. Note that each countable tree is isomorphic to a subtree of $2^{<\omega}$.

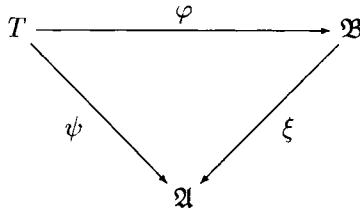
Let $T \subseteq 2^{<\omega}$ be a subtree. A map φ from T to a BA $\mathfrak{B}$ is called *admissible* if the following conditions hold.

- (i) $\varphi(\emptyset) = 1$,
- (ii) $\varphi(\tau) = \varphi(\tau * 0) \cup \varphi(\tau * 1)$, $\tau * 0 \in T$,
- (iii) $\tau \mid \sigma \implies \varphi(\tau) \cap \varphi(\sigma) = 0$, $\tau, \sigma \in T$.

The tree is said to *generate* the BA $\mathfrak{B}$ if there exists an admissible map $\varphi : T \rightarrow \mathfrak{B}$ such that in addition

- (iv) $\varphi(\tau) \neq 0$, $\tau \in T$,
- (v) each element of $\mathfrak{B}$ is a finite join of elements of the type $\varphi(\tau)$.

Lemma 2.1 *Let $T \subset 2^{<\omega}$ be a subtree, and let $\mathfrak{A}$ and $\mathfrak{B}$ be BA's such that T generates $\mathfrak{B}$. Let φ be the corresponding admissible map, and let $\psi : T \rightarrow \mathfrak{A}$ be an admissible map. Then there exists a unique homomorphism ξ such that the following diagram commutes.*



If condition (iv) holds for ψ , then ξ is a monomorphism. If ψ satisfies condition (v), then ξ is an epimorphism.

Now it follows that a BA that is generated by a given tree is unique up to isomorphism. If we have a tree T , we can always construct a BA $\mathfrak{B}$ such that T generates $\mathfrak{B}$. For example, this can be done in the following way.

Let A be a countable set. We define a map $\varphi : T \rightarrow \mathcal{P}(A)$ by induction on the length of strings. If $\varphi(\tau)$ is determined, then $\varphi(\tau)$ is infinite. We divide the set $\varphi(\tau)$ into two infinite parts such that $A_1 \cup A_2 = \varphi(\tau)$ and $A_1 \cap A_2 = \emptyset$, and put $\varphi(\tau * 0) = A_1$ and $\varphi(\tau * 1) = A_2$.

We define

$$\mathfrak{C}(T) \triangleq \text{gr}(\{\varphi(\tau) \mid \tau \in T\}) \subset \mathcal{P}(A).$$

It is clear that conditions (i)–(v) hold for this constructed map $\varphi : T \rightarrow \mathfrak{C}(T)$, and that the isomorphism type of $\mathfrak{C}(T)$ does not depend on the specific construction of φ .

The next assertion reflects some results of this construction.

Lemma 2.2 *Let $T \in \Sigma_{n+1}^A$, then $\mathfrak{C}(T) \in \Delta_{n+1}^A$.*

We have $\Sigma_{n+1}^A = \Sigma_1^{A^{(n)}}$. Therefore, it is sufficient to prove the lemma only for $n = 0$. However, this is almost obvious, because each element of $\mathfrak{C}(T)$ will be enumerated at some stage s , and for enumerated elements, the equality problem is decidable with an appropriate oracle.

Now we describe another way of representing BA's [22]. This method is inspired by the following simple fact.

Lemma 2.3 *If $(\mathfrak{B}, \nu)$ is a numerated BA, then $\mathfrak{B} \simeq \tilde{Q}/\mathfrak{I}$, where $\tilde{Q}$ is a recursive atomless BA, $\mathfrak{I}$ is an ideal of $\tilde{Q}$, and $\mathfrak{I}$ is recursively isomorphic to $\sim_\nu$.*

We define

$$\nu[\sigma] \triangleq \left(\bigcap_{\sigma(i)=1} \nu i \right) \cap \left(\bigcap_{\sigma(i)=0} c\nu i \right) \quad \text{and} \quad A_\nu \triangleq \{\nu[\sigma] \mid \sigma \in 2^{<\omega}\}.$$

Later on, A_ν will be called a standard generating family of $(\mathfrak{B}, \nu)$. It is clear that $\text{gr}(A_\nu) = \mathfrak{B}$.

The tree $2^{<\omega}$ generates a recursive atomless BA $\tilde{Q}$, and the ideal $\mathfrak{I}$ generated by the set $\{\sigma \mid \nu[\sigma] = 0\}$ satisfies the conditions of the lemma.

Let $\{\nu[\sigma] \mid \sigma \in 2^{<\omega}\}$ be a standard generating family of $(\mathfrak{B}, \nu)$. A set $P = \{\sigma \mid \nu[\sigma] \neq 0\}$ is a segment that satisfies the condition

$$\tau \in P \implies (\exists i \leq 1)(\tau * i \in P). \quad (*)$$

Definition 2.1 A *normal segment* is a segment that satisfies condition (*).

Let $P[F] \rightleftharpoons \{\tau \in P \mid \exists \sigma \in F(\sigma \subseteq \tau)\}$, where $P, F \subseteq 2^{<\omega}$. We associate a BA $\text{al}(T)$ with each normal segment T . It is a subalgebra of $\mathcal{P}^*(T)$, consisting of all equivalence classes $T[F]^*$, where F is a finite subset of $2^{<\omega}$. Note that if $(\mathfrak{B}, \nu)$ is a numerated BA and $P = \{\sigma \mid \nu[\sigma] \neq 0\}$, then $\mathfrak{B} \simeq \text{al}(P)$, where isomorphism is defined by the map $T[F]^* \rightarrow \cup_{\sigma \in F} \nu[\sigma]$.

The connection between algorithmic properties of BA's and normal segments is established by the next lemma.

Lemma 2.4 A BA $\mathfrak{B}$ is Σ_1^A - (or Π_1^A -, Δ_1^A -) *constructivizable* if and only if $\mathfrak{B} \simeq \text{al}(T)$ for a suitable normal segment $T \in \Pi_1^A$ (or Σ_1^A , Δ_1^A).

The following sufficient condition on the isomorphism of two BA's $\text{al}(S)$ and $\text{al}(T)$ makes the use of this method of representing BA's very convenient.

Lemma 2.5 Let S and T be normal segments, and φ be an isomorphism from $(S, \subseteq)$ to $(T, \subseteq)$ such that for any $\tau \in T$ there exists $\sigma \in S$ with $\tau \subseteq \varphi(\sigma)$. Then the BA's $\text{al}(S)$ and $\text{al}(T)$ are isomorphic.

The required isomorphism is determined by the map $S[F]^* \mapsto T[\varphi(F)]^*$.

At the end of this section, we establish the connection between the two presentations of BA's that were described above.

For an initial segment P of the tree $2^{<\omega}$, we define

$$T(P) \rightleftharpoons \{\sigma \mid \sigma \in P \ \& \ \sigma = \tau * i \ \& \ \tau * (1 - i) \in P\} \cup \{\emptyset\}.$$

It is clear that the set $T(P)$ is in general not a subtree of the full tree $2^{<\omega}$, but the ordering $\subseteq$ induces the structure of a binary tree on $T(P)$.

Let $\tilde{\mathfrak{C}}(P)$ denote the BA generated by the tree $T(P)$.

Lemma 2.6 For any normal segment $P \subseteq 2^{<\omega}$, the BA's $\tilde{\mathfrak{C}}(P)$ and $\text{al}(P)$ are isomorphic.

The tree $T(P)$ is isomorphic to a suitable subtree $S \subseteq 2^{<\omega}$. We fix some isomorphism $\xi : S \rightarrow T(P)$. Then the map $\psi : S \rightarrow \text{al}(P)$, where $\sigma \mapsto P[\{\xi(\sigma)\}]^*$, induces the required isomorphism (see Lemma 2.1).

3 Arithmetical hierarchy of Boolean algebras

In this section we study the structure of the arithmetical hierarchy of BA's and the methods of representing arithmetical BA's as factor algebras of constructive BA's with respect to some natural ideals.

Theorem 3.1 (Odintsov [27]) *Any Π_1^A -constructivizable BA is Δ_1^A -constructivizable.*

Let $\mathfrak{B}$ be a Π_1^A -constructivizable BA. By Lemma 2.4, $\mathfrak{B} \simeq \text{al}(\mathcal{D})$, where $\mathcal{D}$ is a Σ_1^A -normal segment. Let $\{\mathcal{D}_s \mid s \in \omega\}$ be an effective approximation of $\mathcal{D}$ with an oracle A by finite initial segments of the tree $2^{<\omega}$, where $|\mathcal{D}^{s+1} \setminus \mathcal{D}^s| \leq 1$, $s \in \omega$.

We construct the required A -constructive BA $\mathfrak{A}$, with $\mathfrak{A} \simeq \mathfrak{B}$, in stages. Assume that we have a finite BA $\mathfrak{A}^s$ and an isomorphism $\varphi_s : \tilde{\mathfrak{C}}(\mathcal{D}^s) \rightarrow \mathfrak{A}^s$ at the end of stage s , and $T(\mathcal{D}^s) \neq T(\mathcal{D}^{s+1})$ holds. Only one pair of elements can be added to the tree $T(\mathcal{D}^s)$ at each stage, because $|\mathcal{D}^{s+1} \setminus \mathcal{D}^s| \leq 1$. We add $\sigma * 0$ and $\sigma * 1$, and let $\tau \equiv \bigcup \{\rho \mid \rho \subseteq \sigma \ \& \ \rho \in T(\mathcal{D}^s)\}$. The number of atoms lying below τ in $\mathfrak{C}(\mathcal{D}^{s+1})$ is one plus the number of atoms lying below τ in $\tilde{\mathfrak{C}}(\mathcal{D}^s)$. Therefore we divide one of the atoms of $\mathfrak{A}^s$ below $\varphi_s(\tau)$ into two parts to obtain $\mathfrak{A}^{s+1}$. Let

$$\xi^{s+1} : [\tau]_{\tilde{\mathfrak{C}}(\mathcal{D}^{s+1})} \longrightarrow [\varphi(\tau)]_{\mathfrak{A}^{s+1}}$$

be an isomorphism. We can extend this isomorphism to obtain a new isomorphism $\varphi^{s+1} : \tilde{\mathfrak{C}}(\mathcal{D}^{s+1}) \rightarrow \mathfrak{A}^{s+1}$ such that the following condition holds.

$$\sigma \in T(\mathcal{D}^s) \ \& \ \neg(\sigma \subseteq \tau) \implies \varphi^{s+1}(\sigma) = \varphi^s(\sigma)$$

This construction is effective with the oracle A , therefore $\mathfrak{A} \equiv \bigcup_{s \in \omega} \mathfrak{A}^s$ is an A -constructivizable BA. The isomorphism $\mathfrak{A} \simeq \mathfrak{B}$ is easily established. It is sufficient to note that the sequence of isomorphisms $\{\varphi_s\}$ is stabilized on each element of $T(\mathcal{D})$.

Corollary 3.2 (Dzgoev [17]) *If a BA $\mathfrak{B}$ is constructivizable as a partial order, then it is constructivizable as a Boolean algebra.*

Any constructive presentation of the partial order $(|\mathfrak{B}|, \leq_{\mathfrak{B}})$ is a Π_1 -constructive presentation of the BA $\mathfrak{B}$, because the operations of the BA are definable by Π_1 -sentences of a first order language.

Theorem 3.3 (Selivanov [27]) *Any Δ_2^A -constructivizable BA is Σ_1^A -constructivizable.*

By Lemma 2.4, it is sufficient to check that for any normal segment $\mathcal{D} \in \Delta_2^A$, there exists a normal segment $T \in \Pi_1^A$ such that $\text{al}(T) \simeq \text{al}(\mathcal{D})$.

We define $\varphi : \mathcal{D} \rightarrow 2^{<\omega}$. Let $\varphi(\emptyset) = \emptyset$, and assume that $\varphi(\sigma)$ is already defined. If $\sigma * 0, \sigma * 1 \in \mathcal{D}$, then $\varphi(\sigma * i) = \varphi(\sigma) * i$, $i \leq 1$. If $\sigma * k \in \mathcal{D}$ and $\sigma * (1 - k) \notin \mathcal{D}$, then $\varphi(\sigma * k) = \varphi(\sigma) * 0$. It is clear that $\varphi(\mathcal{D}) \in \Delta_2^A$, and φ is an isomorphism from $(\mathcal{D}; \subseteq)$ to $(\varphi(\mathcal{D}); \subseteq)$; and if $\tau \in \varphi(\mathcal{D})$, then $\tau * 0 \in \varphi(\mathcal{D})$. So, without loss of generality, we can assume that $\mathcal{D}$ has the following property.

$$\sigma \in \mathcal{D} \implies \sigma * 0 \in \mathcal{D} \quad (*)$$

Let $\{\mathcal{D}^s\}_{s \in \omega}$ be a Δ_2^A -approximation of $\mathcal{D}$ by finite initial segments of $2^{<\omega}$, i.e.,

$$(\sigma \in \mathcal{D} \iff (\exists s \forall t > s \sigma \in \mathcal{D}^t)) \ \& \ (\sigma \notin \mathcal{D} \iff (\exists s \forall t > s \sigma \notin \mathcal{D}^t)).$$

Moreover, we suggest that this approximation has the property that, for any s ,

$$(|\mathcal{D}^{s+1} \setminus \mathcal{D}^s| = 1 \ \& \ \mathcal{D}^s \subset \mathcal{D}^{s+1}) \vee (\mathcal{D}^{s+1} \subseteq \mathcal{D}^s).$$

Now we give a sketch of the construction. At each stage s we shall construct a finite isomorphism $\psi^s : \mathcal{D}^s \rightarrow 2^{<\omega}$.

CASE 1: $\tau \in \mathcal{D}^{s+1} \setminus \mathcal{D}^s$, $\tau = \sigma * i$. We have $|\mathcal{D}^{s+1} \setminus \mathcal{D}^s| = 1$, therefore $\sigma \in \mathcal{D}^s$. Let $\rho = \psi^s(\sigma) * 0^k$ be a maximal element of $\psi^s(\mathcal{D}^s)$. We define $\psi^{s+1}(\sigma * \sigma_0) = \rho * \sigma_0$, where $\sigma * \sigma_0 \in \mathcal{D}^{s+1}$. In the other cases, ψ^{s+1} is identical with ψ^s .

CASE 2: $\mathcal{D}^{s+1} \subseteq \mathcal{D}^s$. We put $\psi^{s+1} = \psi^s \upharpoonright \mathcal{D}^s$. Let $\psi = \lim_s \psi^s$,

$$T \simeq \{\sigma \mid \exists \tau (\sigma \subset \psi(\tau))\}.$$

By Lemma 2.5, we have $\text{al}(\mathcal{D}) \simeq \text{al}(T)$. To finish the proof, we have to establish that T is a Π_1^A -normal segment. This follows from the equivalence:

$$\sigma \in 2^{<\omega} \setminus T \iff$$

$$\exists s, \tau, \rho ((\tau * 0 \subset \psi^s(\rho)) \ \& \ (\tau * 1 \in \psi^s(\mathcal{D}^s)) \ \& \ (\tau * 1 \subset \sigma)).$$

Here we need to use condition (*).

The next statement follows from Theorems 3.1 and 3.3.

Corollary 3.4 *Any Π_2^A -constructivizable BA is Σ_1^A -constructivizable.*

So we have established that the arithmetical hierarchy of BA's has the following structure.

$$\begin{array}{ccccccc} \Delta_1^A & \subset & \Sigma_1^A & \subset & \Sigma_2^A & \subset & \dots \\ \parallel & & \parallel & & \parallel & & \\ \Pi_1^A & & \Delta_2^A & & \Delta_3^A & & \dots \\ & & \parallel & & \parallel & & \\ & & \Pi_2^A & & \Pi_3^A & & \dots \end{array}$$

All inclusions in this diagram are proper. This follows from the classical result of Feiner [12]. Let Σ_α and Π_α , where α is a recursive ordinal, be the classes of the hyperarithmetical hierarchy.

Theorem 3.5 (Feiner [12]) *For any α there exists a $\Sigma_{\alpha+1}$ -constructivizable BA which is not Σ_α -constructivizable.*

This result cannot be refined in terms of inclusions of classes of Σ_α - and Π_α -constructivizable BA's, because any $\Pi_{\alpha+1}$ -constructivizable BA is $\Delta_{\alpha+1}$ -constructivizable by Theorem 3.1, and any $\Pi_{\alpha+2}$ -constructivizable BA is $\Sigma_{\alpha+1}$ -constructivizable by Corollary 3.4. In particular, we cannot construct refinements of the arithmetical hierarchy of BA's. For example, any Σ_n^{-1} -constructivizable BA is a Δ_1^0 -constructivizable BA, where Σ_n^{-1} are the classes of Ershov's hierarchy, see [8, 10].

In the second part of this section we show that all arithmetical BA's can be presented as factor algebras of constructive BA's with respect to certain natural ideals. Simultaneously, we can consider these results as an exact estimation of the algorithmic complexity of factorizations of numerated BA's by certain natural ideals.

Theorem 3.6 [27]

- (i) (Selivanov) *Any Σ_2^A -constructivizable BA is isomorphic to a factorization of a suitable atomic Δ_1^A -constructivizable BA with respect to its Fréchet ideal.*
- (ii) (Odintsov) *Any Σ_1^A -constructivizable BA is isomorphic to a factorization of a suitable atomic A -strongly constructivizable BA with respect to its Fréchet ideal.*

We give a proof for the first part of the theorem, omitting the condition “to be atomic”.

Let $\mathfrak{B}$ be a Σ_2^A -constructive BA, and P be the corresponding Π_2^A -normal segment, such that $\mathfrak{B} \simeq \text{al}(P)$. Then there exists a function $f : 2^{<\omega} \times \omega \rightarrow \omega$ that is recursive with respect to A , such that $f(\sigma, s) \leq f(\sigma, s+1)$ and $\sigma \in P \iff \lim_s f(\sigma, s) = \omega$. We define a Σ_1^A -subtree $\mathcal{D}$ of $2^{<\omega}$ in the following way.

$$\sigma * i \in \mathcal{D} \iff \exists s \forall \tau \subset \sigma (f(\tau, s) \geq \text{lh}(\sigma)).$$

Let us consider the BA $\mathfrak{C}(\mathcal{D})$. We have $\mathfrak{B} \simeq \mathfrak{C}(\mathcal{D})/\text{F}(\mathfrak{C}(\mathcal{D}))$. To prove this fact, it is sufficient to note that

$$\sigma \notin P \iff \lim_s f(\sigma, s) < \omega \iff |\{\tau \in \mathcal{D} \mid \sigma \subset \tau\}| < \omega.$$

This last condition implies that the elements of $\mathfrak{C}(\mathcal{D})$ corresponding to the string τ is an element of the Fréchet ideal. Therefore the identity map of $2^{<\omega}$ onto itself induces an isomorphism of $\mathfrak{B}$ and $\mathfrak{C}(\mathcal{D})/\text{F}(\mathfrak{C}(\mathcal{D}))$. The BA $\mathfrak{C}(\mathcal{D})$ is A -constructivizable by Lemma 2.2.

To make the BA $\mathfrak{C}(\mathcal{D})$ atomic, we need a slight modification of our construction. We should add one atom under each element of the constructing tree. This does not change the isomorphism type of the factorization by the Fréchet ideal.

The second part of this theorem has an analogous, but simpler, proof. We merely remark that the strong constructivity of an atomic, constructive BA is equivalent to the decidability of the set of atoms (see Theorem 1.5).

Repeated applications of Theorems 3.3 and 3.6 yield:

Corollary 3.7 *Let $n < \omega$. Any Σ_{2n+2} - (or Σ_{2n+1} -) constructivizable BA is isomorphic to a factorization of a suitable n -atomic constructivizable (or strongly constructivizable) BA by the ideal F_{n+1} .*

It follows from the definition of the Fréchet ideal, that if $(\mathfrak{B}, \nu)$ is a constructive (or strongly constructive) BA, then $\mathfrak{B}/\text{F}_{n+1}(\mathfrak{B})$ is Σ_{2n+2} - (or Σ_{2n+1} -) constructivizable. Therefore $\mathfrak{B} \mapsto \mathfrak{B}/\text{F}_{n+1}(\mathfrak{B})$ is a surjective map of the class of all constructivizable (or strongly constructivizable) BA's to the class of all Σ_{2n+2} - (or Σ_{2n+1} -) constructivizable BA's, and this map is one-to-one on the class of n -atomic BA's [15].

Let $\text{FAtl}(\mathfrak{B})$ denote the ideal generated by the atoms and the atomless elements of a BA $\mathfrak{B}$. We conclude this section with an analogue of Theorem 3.6 for the ideals $\text{FAtl}(\mathfrak{B})$ and $I(\mathfrak{B})$ (the Ershov-Tarski ideal). However, first we have to define an operation on trees.

For $\sigma \in 2^{<\omega}$ and $T \subseteq 2^{<\omega}$, we put $\sigma * T \Leftrightarrow \{\sigma * \tau \mid \tau \in T\}$ and $T(\sigma) \Leftrightarrow \{\tau \mid \sigma * \tau \in T\}$. We define an operation which takes a sequence of segments $\{T_\tau\}_{\tau \in 2^{<\omega}}$, and constructs a new segment $\prod_\tau T_\tau$ in the following way.

Let W be the set of all strings which have zeros on the even places. We put $\tilde{\sigma} = \sigma$ and $\sigma * i = \tilde{\sigma} * 0i$. The map $\sigma \mapsto \tilde{\sigma}$ is an isomorphism from $(2^{<\omega}; \subseteq)$ to $(W; \subseteq)$, and for any $\tau \in W$, there exists σ such that $\tau \subseteq \tilde{\sigma}$. It is clear that $\{\tilde{\sigma} * 1 \mid \sigma \in 2^{<\omega}\}$ is the set of all maximal elements of $(2^{<\omega} \setminus W; \subseteq)$, and that $W = \{\tilde{\sigma}, \tilde{\sigma} * 0 \mid \sigma \in 2^{<\omega}\}$. If $\sigma \in W \setminus \{\tilde{\tau} \mid \tau \in 2^{<\omega}\}$, then $\sigma = \tilde{\tau} * 0$ for some $\tau \in 2^{<\omega}$, thus $\sigma * 0 = \tau * 0$, and $\sigma * 1 = \tau * 1$. Now we put

$$\prod_\tau T_\tau \Leftrightarrow W \cup (\bigcup_\tau \tilde{\tau} * 1 * T_\tau).$$

The next two lemmata state some simple properties of this operation.

Lemma 3.8 *Let $T = \prod_\tau T_\tau$.*

- (i) *If $\{T_\tau\}$ is a Σ_1^A -sequence of trees, then T is a Σ_1^A -tree.*
- (ii) *For any $\sigma \in 2^{<\omega}$, we have $T(\tilde{\sigma} * 1) = T_\sigma$ and $T(\tilde{\sigma}) = \prod_\tau T_{\sigma * \tau}$.*

From now on, we only consider non-empty trees.

Lemma 3.9 *Let $\{T_\tau\}$ be a sequence of trees, $\mathfrak{B}$ a BA generated by the tree $T \Leftrightarrow \prod_\tau T_\tau$, and b_τ an element of $\mathfrak{B}$ corresponding to $\tau \in T$.*

- (i) *If all elements $b_{\tilde{\sigma} * 1}$ are atomic, then $\mathfrak{B}$ is atomic.*
- (ii) *If all elements $b_{\tilde{\sigma} * 1}$ lie in $\text{FAtl}(\mathfrak{B})$, and almost all of these elements are atomless, then $1_{\mathfrak{B}} \in \text{FAtl}(\mathfrak{B})$.*
- (iii) *If all elements $b_{\tilde{\sigma} * 1}$ lie in $I(\mathfrak{B})$, and almost all of these elements are atomic, then $1_{\mathfrak{B}} \in I(\mathfrak{B})$.*
- (iv) *If infinitely many elements of the type $b_{\tilde{\sigma} * 1}$ lie in $I(\mathfrak{B})$ are neither atomic nor atomless, then $1_{\mathfrak{B}} \notin I(\mathfrak{B})$.*

Theorem 3.10 (Odintsov [27]) *Any Σ_3^A -constructivizable BA is isomorphic to a factor algebra of a suitable Δ_1^A -constructivizable BA by the ideal FAtI .*

Let $(\mathfrak{A}, \nu)$ be a Σ_3^A -constructive BA, then there exists a Σ_1^A -sequence $\{\psi_\tau \mid \tau \in 2^{<\omega}\}$ of non-empty initial segments of ω such that

$$\nu[\sigma] = 0 \iff (\text{for almost all extensions } \tau \supset \sigma, \psi_\tau \text{ is equal to } \omega),$$

where $\{\nu[\sigma] \mid \sigma \in 2^{<\omega}\}$ is a standard generating family of the numerated BA $(\mathfrak{A}, \nu)$, see [22, §4].

The sequence of trees $\{R_\tau \mid \tau \in 2^{<\omega}\}$, where $R_\tau = \{\sigma \mid \text{lh}(\sigma) \in \psi_\tau\}$ is a Σ_1^A -sequence, therefore $S \simeq \prod_\tau R_\tau$ is a Σ_1^A -tree, by Lemma 3.8 (i). Let us consider $\mathfrak{B} \simeq \mathfrak{C}(S)$. We have $\mathfrak{B}/\text{FAtI}(\mathfrak{B}) \simeq \mathfrak{A}$. Indeed, $[b_{\tilde{\sigma}*1}]_{\mathfrak{B}}$ is a finite BA if $|\psi_\sigma| < \omega$, and atomless if $|\psi_\sigma| = \omega$. In any case, we have $b_{\tilde{\sigma}*1} \in \text{FAtI}(\mathfrak{B})$. Therefore the family $\{b_{\tilde{\sigma}}/\text{FAtI}(\mathfrak{B}) \mid \sigma \in 2^{<\omega}\}$ generates $\mathfrak{B}/\text{FAtI}(\mathfrak{B})$. By Lemma 3.10 (ii), we can conclude that

$$b_{\tilde{\sigma}} \in \text{FAtI}(\mathfrak{B}) \iff (|\psi_\tau| = \omega \text{ for almost all } \tau \supset \sigma).$$

Hence, the identity map of the tree $2^{<\omega}$ onto itself induces an isomorphism between $\mathfrak{B}/\text{FAtI}(\mathfrak{B})$ and $\mathfrak{A} = \text{gr}(\{\nu[\sigma] \mid \sigma \in 2^{<\omega}\})$.

Via a simple refinement of this construction, we obtain:

Theorem 3.11 (Selivanov [27]) *Any Σ_4^A -constructivizable BA is isomorphic to the factor algebra of a suitable Δ_1^A -constructivizable BA by the Ershov-Tarski ideal.*

The iterated ideals I_n and FAtI_n , $n < \omega$, are defined in the usual way.

Corollary 3.12 *Any Σ_{3n+3} -constructivizable BA is isomorphic to the factor algebra of a suitable constructive BA by the ideal FAtI_{n+1} .*

Corollary 3.13 *Any Σ_{4n+4} -constructivizable BA is isomorphic to the factor algebra of a suitable constructive BA by the ideal I_{n+1} .*

It is easy to see that in Theorems 3.1, 3.3, 3.6, 3.10 and 3.11, BA's are effectively constructed. Thus, for instance, the following assertion is true: for any Σ_{4n+1} -constructive BA $(\mathfrak{A}, \nu)$ we can effectively find a Σ_1 -constructive BA $(\mathfrak{B}, \mu)$ such that $\mathfrak{B}/I_n(\mathfrak{B}) \simeq \mathfrak{A}$, and so on.

We finish this section with the following open question. What is the structure of the numerated BA's that lie in the limit classes of the hyperarithmetical hierarchy, i.e., in the classes Σ_α , where α is a limit ordinal?

4 Restricted theories of constructive Boolean algebras

This section is devoted to the study of n -constructivity of BA's for various n . First of all, we introduce the general notions that we need in this section. Let $\mathcal{F}_n$ be the set of all first order formulae which have $\leq n$ alternations of quantifiers in prenex normal form. We suppose that some algorithm for reducing to prenex normal form is fixed. Let $(\mathfrak{M}, \nu)$ be a numerated model. Then

$$\text{Th}_n(\mathfrak{M}, \nu) \rightleftharpoons \{ \langle m, \ell_0, \dots, \ell_{k_m-1} \rangle \mid \mathfrak{M} \models \varphi_m(\nu \ell_0, \dots, \nu \ell_{k_m-1}) \ \& \ \varphi_m \in \mathcal{F}_n \}$$

$$\text{Th}(\mathfrak{M}, \nu) \rightleftharpoons \bigcup_{n < \omega} \text{Th}_n(\mathfrak{M}, \nu)$$

Recall that $\{\varphi_m(v_0, \dots, v_{k_m-1})\}_{m < \omega}$ is a Gödel numeration of all first order formulae. Of course, the sets $\mathcal{F}_n$, $n < \omega$, are uniformly decidable with respect to this numeration.

A numerated model $(\mathfrak{M}, \nu)$ is said to be n -constructive if $\text{Th}_n(\mathfrak{M}, \nu)$ is recursive. A model $\mathfrak{M}$ is n -constructivizable if there exists a numeration ν of $\mathfrak{M}$ such that $(\mathfrak{M}, \nu)$ is n -constructive. $\text{Th}_n(\mathfrak{M}, \nu)$ is called an n -restricted theory of a numerated model $(\mathfrak{M}, \nu)$. It is obvious that the decidability of $\text{Th}(\mathfrak{M}, \nu)$ is equivalent to the strong constructivity of $(\mathfrak{M}, \nu)$, and the decidability of $\text{Th}_0(\mathfrak{M}, \nu)$ is equivalent to the constructivity of $(\mathfrak{M}, \nu)$.

With the help of Theorem 1.5, we can obtain several sufficient conditions for the strong constructivity of a numerated model which has a decidable n -restricted theory for some n . These conditions are summarized in the following proposition.

Proposition 4.1 *Let $(\mathfrak{B}, \nu)$ be a numerated BA.*

- (i) *If $\text{ch}_1(\mathfrak{B}) = n$ and $(\mathfrak{B}, \nu)$ is $(4n + 3)$ -constructive, then $(\mathfrak{B}, \nu)$ is strongly constructive.*
- (ii) *If $\text{ch}(\mathfrak{B}) = (n, \omega, 0)$ and $(\mathfrak{B}, \nu)$ is $(4n + 1)$ -constructive, then $(\mathfrak{B}, \nu)$ is strongly constructive.*
- (iii) *If $\text{ch}_1(\mathfrak{B}) = (n, 0, 1)$ and $(\mathfrak{B}, \nu)$ is $(4n)$ -constructive, then $(\mathfrak{B}, \nu)$ is strongly constructive.*

- (iv) If $\text{ch}_1(\mathfrak{B}) = (n+1, 1, 0)$ and $(\mathfrak{B}, \nu)$ is $(4n+3)$ -constructive, then $(\mathfrak{B}, \nu)$ is strongly constructive.

The following question arises at this point: is it possible to decrease the levels of decidability of the restricted theories mentioned in Proposition 4.1 that are sufficient for the strong constructivity of a BA?

The first result in this direction was obtained by Goncharov [15]. Using the technique of Feiner hierarchies [12], he proved the following:

Theorem 4.2 *There exists an atomic constructivizable BA that is not strongly constructivizable.*

It is clear that the BA mentioned in this theorem is infinite, i.e., the elementary characteristic of this BA is $(0, \omega, 0)$. This means that Goncharov constructed a BA that is 0-constructivizable, but not 1-constructivizable (see Proposition 4.1 (ii)). We will not discuss the method suggested by Feiner, nor the various aspects of its application. The details can be found in [12, 15, 33]. The result mentioned above admits a generalization (see [16]).

Theorem 4.3 *For any $n \geq 0$, there exists a $(4n)$ -constructivizable BA that is not $(4n+1)$ -constructivizable.*

We will give a sketch of the proof of this theorem, but first we say a few words about the connection of BA's and linear orders.

If L is a linear order with a first element, let $\mathfrak{B}_L$ denote the interval BA of this order, i.e., $\mathfrak{B}_L$ is the subalgebra of $\mathcal{P}(L)$ generated by the sets $\{[a, b) \mid a, b \in L\}$, where $[a, b) = \{x \in L \mid a \leq x < b\}$.

The following two propositions establish the connection between algorithmic properties of BA's and linear orders.

Proposition 4.4 *If $\mathbb{L} = (L, \nu)$ is a numerated, constructive linear order, then the BA $\mathfrak{B}_L$ has a numeration $\gamma_{\mathbb{L}}$ such that $(\mathfrak{B}_L, \gamma_{\mathbb{L}})$ is constructive.*

So we have a preferred method of constructing a numeration $\gamma_{\mathbb{L}}$ of $\mathfrak{B}_L$ for any presentation (L, ν) of a linear order L .

Proposition 4.5 *For any constructive BA $(\mathfrak{B}, \nu)$, there exists a constructive linear order $\mathbb{L} = (L, \mu)$ such that $(\mathfrak{B}, \nu)$ and $(\mathfrak{B}_L, \gamma_{\mathbb{L}})$, are recursively isomorphic.*

Both results admit a natural relativization, but we only need the simplest case.

If $L_0 = (|L_0|; <_0)$ and $L_1 = (|L_1|; <_1)$ are linear orders with $|L_0| \cap |L_1| = \emptyset$, then the orders $L_0 + L_1$ and $L_0 \times L_1$ are defined in the following way.

$$|L_0 + L_1| = |L_0| \cup |L_1|,$$

$$x <_+ y \Leftrightarrow (x \in L_0 \ \& \ y \in L_1) \vee (x \in L_0 \ \& \ y \in L_0 \ \& \ x <_0 y) \\ \vee (x \in L_1 \ \& \ y \in L_1 \ \& \ x <_1 y);$$

$$|L_0 \times L_1| = |L_0| \times |L_1|,$$

$$(x_0, y_0) <_\times (x_1, y_1) \Leftrightarrow (y_0 <_1 y_1) \vee (y_0 = y_1 \ \& \ x_0 <_0 x_1).$$

The finite sum and product of more than two linear orders, and the infinite sum of linear orders can be defined in the usual way. Let ω , η , and $\bar{n}$ denote the order types of the natural numbers $\mathbb{N}$, the rational numbers $\mathbb{Q}$, and the number set $\{0, \dots, n-1\}$ with the usual ordering, respectively. Then we define the linear orders L_n , L_n^0 , and L_n^1 , $n < \omega$, by induction on n :

$$L_0 = \bar{2}, \quad L_0^0 = \omega, \quad L_0^1 = \bar{1} + \eta,$$

$$L_1 = (\omega + \eta) \times \omega,$$

$$L_{n+1} = (L_n^0 + L_n^1) \times \omega, \quad L_{n+1}^0 = L_n \times \omega, \quad L_{n+1}^1 = \bar{1} + (L_n^0 + L_n^1) \times \eta$$

Let us consider the BA's $\mathfrak{B}_n \Leftrightarrow \mathfrak{B}_{L_n}$, $\mathfrak{B}_n^0 \Leftrightarrow \mathfrak{B}_{L_n^0}$, and $\mathfrak{B}_n^1 \Leftrightarrow \mathfrak{B}_{L_n^1}$. It is easy to show that the BA $\mathfrak{B}_\eta$ is atomless, the BA $\mathfrak{B}_\omega$ is atomic, and that the following fact is true.

Lemma 4.6 *Let L and L_1 be linear orders such that the BA $\mathfrak{B}_L$ is atomic, and let $\mathfrak{B}' = \mathfrak{B}_{(L+\bar{1}+\eta) \times \omega \times L_1}$. Then $\mathfrak{B}'/I(\mathfrak{B}') \simeq \mathfrak{B}_{L_1}$.*

With the help of this lemma, we can calculate the elementary characteristics of $\mathfrak{B}_n$, $\mathfrak{B}_n^0$, and $\mathfrak{B}_n^1$, $n < \omega$:

$$\text{ch}(\mathfrak{B}_n) = (n, 1, 0), \quad \text{ch}(\mathfrak{B}_n^0) = (n, \omega, 0), \quad \text{ch}(\mathfrak{B}_n^1) = (n, 0, 1).$$

The constructive character of addition and multiplication of linear orders allows us to prove the following.

Proposition 4.7 *There exist numerations ν_n, ν_n^0 and ν_n^1 such that the numerated BA's $(\mathfrak{B}_n, \nu_n)$, $(\mathfrak{B}_n^0, \nu_n^0)$ and $(\mathfrak{B}_n^1, \nu_n^1)$ are strongly constructive, and for each of these BA's there exists an algorithm which calculates the elementary characteristic of an element with a given number.*

Another important feature of the BA $\mathfrak{B}_n$ is that $\text{ch}_2(a) < \omega$ for any $a \in \mathfrak{B}_n$. This allows us to prove the following fact.

Proposition 4.8 *Let $\mathfrak{B}$ be a BA. If $\text{ch}_2(\mathfrak{B}) < \omega$, then there exists a BA $\mathfrak{B}'$ such that $\mathfrak{B}' \equiv \mathfrak{B}$ and $(\forall a)(a \in |\mathfrak{B}'| \implies \text{ch}_2(a) < \omega)$.*

If $\text{ch}_2(\mathfrak{B}) = \omega$, then there exists a BA $\mathfrak{B}'$ such that $\mathfrak{B}' \equiv \mathfrak{B}$, and

$$(\forall a) \left(a \in |\mathfrak{B}'| \implies ((\text{ch}_1(a) < n_1 \implies \text{ch}_2(a) < \omega) \right.$$

$$\left. \& (\text{ch}_1(a) < n_1 \vee (\text{ch}_1(a) = n_1 \& (\text{ch}_2(a) < \omega \vee (\text{ch}_1(c(a)) = n_1 \right.$$

$$\left. \& \text{ch}_2(c(a)) < \omega) \vee \text{ch}_2(c(a)) < n_1) \right) \Big),$$

where $n_1 < \text{ch}_2(b)$.

This fact is important because, for any triple (m_1, m_2, m_3) with $m_1, m_2 < \omega$ and $m_3 \in \{0, 1\}$, the property “to have the elementary characteristic (m_1, m_2, m_3) ” can be expressed with only one first order formula.

The proofs of the next two propositions use a technique developed by Fraïssé [13], Ehrenfeucht [6] and Taimanov [38]. Both assertions have been proved in [16].

Proposition 4.9 *Let $\mathfrak{B}_1$ and $\mathfrak{B}_2$ be BA's, and let $a_1, \dots, a_k \in |\mathfrak{B}_1|$, $b_1, \dots, b_k \in |\mathfrak{B}_2|$, where $\&_{i \neq j} a_i \cap a_j = 0$, $\&_{i \neq j} b_i \cap b_j = 0$, $\cup_{i=1}^k a_i = 1$ and $\cup_{i=1}^k b_i = 1$. If $[a_i]_{\mathfrak{B}_1} \equiv_n [b_i]_{\mathfrak{B}_2}$ for any i , then*

$$(\mathfrak{B}_1, a_1, \dots, a_k) \equiv_n (\mathfrak{B}_2, b_1, \dots, b_k).$$

Proposition 4.10 *If $\mathfrak{B}_1$ and $\mathfrak{B}_2$ are BA's such that $\text{ch}_1(\mathfrak{B}_1) > k$ and $\text{ch}_1(\mathfrak{B}_2) > k$, then $\mathfrak{B}_1 \equiv_{4k} \mathfrak{B}_2$.*

Let us consider the BA $\mathfrak{B}$ from Theorem 4.2. By Proposition 4.5, the BA $\mathfrak{B}$ can be presented as an interval algebra $\mathfrak{B}_L$, i.e., $\mathfrak{B} \simeq \mathfrak{B}_L$, where L is a

recursive linear order. Recall that $\mathfrak{B}$ is atomic and $\text{Th}_1(\mathfrak{B}, \nu)$ is undecidable for any numeration ν of $\mathfrak{B}$. Let L_n , $n < \omega$, be the linear orders defined above. We define a new linear order L^n by $L^n \rightleftharpoons L_n \times L$, $n < \omega$. It is clear that $\mathfrak{B}_{L^n}/I_n(\mathfrak{B}_{L^n}) \simeq \mathfrak{B}_L$ and $\text{ch}(\mathfrak{B}_{L^n}) = (n, \omega, 0)$.

Propositions 4.9 and 4.10, and the properties of the BA's $\mathfrak{B}_n$ stated above allow us to prove the following.

Proposition 4.11 *There exists a numeration ν_{L^n} of the BA $\mathfrak{B}_{L^n}$ such that $(\mathfrak{B}_{L^n}, \nu_{L^n})$ is constructive, and for any $a_1, \dots, a_k \in \omega$ we can effectively find $b_1, \dots, b_k \in \omega$ such that*

$$(\mathfrak{B}_{L^n}, \nu_{L^n} a_1, \dots, \nu_{L^n} a_k) \equiv_{4n} (\mathfrak{B}_{n+1}, \nu_{n+1} b_1, \dots, \nu_{n+1} b_k).$$

Now we can conclude the proof of Theorem 4.3. From Proposition 4.11 and the strong constructivity of $(\mathfrak{B}_{n+1}, \nu_{n+1})$, we obtain that $\text{Th}_{4n}(\mathfrak{B}_{L^n}, \nu_{L^n})$ is decidable, i.e., $\mathfrak{B}_{L^n}$ is $4n$ -constructivizable. However, the BA $\mathfrak{B}_{L^n}$ is not $(4n+1)$ -constructivizable. Indeed, if ν is a numeration of $\mathfrak{B}_{L^n}$ such that $\text{Th}_{4n+1}(\mathfrak{B}_{L^n}, \nu)$ is decidable, then

$$\text{Th}_1(\mathfrak{B}_{L^n}/I_n(\mathfrak{B}_{L^n}), \nu/I_n) = \text{Th}_1(\mathfrak{B}_L, \nu')$$

is decidable. This is a contradiction to the properties of $\mathfrak{B}_L$ (see Theorem 4.2).

The above constructed BA $\mathfrak{B}_{L^n}$ has elementary characteristic $(n, \omega, 0)$. Generalizing this construction, we can obtain the following fact.

Theorem 4.12 *For any n and any triple (n_1, n_2, n_3) where $n_3 \in \{0, 1\}$, $n_1, n_2 \leq \omega$ and $(n_1 > n) \vee (n_1 = n \ \& \ n_2 = \omega)$, there exists a BA $\mathfrak{B}$ such that $\text{ch}(\mathfrak{B}) = (n_1, n_2, n_3)$ and $\mathfrak{B}$ is $(4n)$ -constructivizable, but not $(4n+1)$ -constructivizable.*

The proof of the next theorem uses the same technique.

Theorem 4.13 *There exists a BA $\mathfrak{B}$ with elementary characteristic $(\omega, 0, 0)$ such that $\mathfrak{B}$ is not strongly constructivizable, yet there exists a numeration ν of $\mathfrak{B}$ such that $\text{Th}_n(\mathfrak{B}, \nu)$ is decidable for any n .*

Of course, the theories $\text{Th}_n(\mathfrak{B}, \nu)$ cannot be uniformly decidable, because this would be equivalent to the strong constructivity of $(\mathfrak{B}, \nu)$.

The construction of $\mathfrak{B}$ is based on the following observation.

Lemma 4.14 *If $(\mathfrak{B}, \nu)$ is a strongly constructive BA. then*

$$\omega_{\mathfrak{B}} \rightleftharpoons \{m \mid (\exists x \in |\mathfrak{B}|) \text{ ch}(x) = (m, \omega, 0)\} \in \Sigma_2^0.$$

The proof is straightforward, and uses the Tarski-Kuratowski algorithm.

Let us consider $A \in \Pi_2^0 \setminus \Sigma_2^0$. By the algorithm of Kreisel, Schoenfield and Wang Hao [35], we know that there exists a recursive predicate $P(x, y)$ such that $x \in A \iff \exists^\omega y P(x, y)$.

Now we construct a recursive linear order L_A such that $L_A \simeq \sum_{m \in \omega} \Delta_m$, where

$$\Delta_m = \begin{cases} L_m \times \bar{k} & \text{with } k = |\{y \mid P(m, y)\}|, \text{ if } m \notin A, \\ L_m \times \omega = L_{m+1}^0 & \text{if } m \in A. \end{cases}$$

The BA $\mathfrak{B} \rightleftharpoons \mathfrak{B}_{L_A}$ satisfies the condition $\omega_{\mathfrak{B}} = A$, therefore $\mathfrak{B}$ is not strongly constructivizable.

To prove the decidability of the restricted theories $\text{Th}_n(\mathfrak{B}, \nu)$ for a suitable numeration ν , one relies on the properties of a certain recursive presentation of order L_A , and uses the same technique as in the proof of Theorem 4.3.

Let us consider a BA $\mathfrak{B}$ with $\text{ch}(\mathfrak{B}) = (1, 1, 0)$. For any element of this BA, either the element itself or its complement lies in the ideal $I(\mathfrak{B})$. Therefore, for any constructivization ν of $\mathfrak{B}$, the strong constructivity of $(\mathfrak{B}, \nu)$ is equivalent to the decidability of the sets $A(\mathfrak{B})$, $\text{Atl}(\mathfrak{B})$ and $\text{At}(\mathfrak{B})$ with respect to ν . However, this sufficient condition on strong constructivity can be weakened (see [26]).

Theorem 4.15 *If $(\mathfrak{B}, \nu)$ is a constructive BA, $\text{ch}(\mathfrak{B}) = (1, 1, 0)$, and the sets $\nu^{-1}(A(\mathfrak{B}))$ and $\nu^{-1}(\text{Atl}(\mathfrak{B}))$ are recursive, then there exists a numeration μ of $\mathfrak{B}$ such that $(\mathfrak{B}, \mu)$ is strongly constructive.*

The natural sufficient condition on strong constructivizability of BA's with elementary characteristic $(1, 0, 1)$ can be weakened as well. By Proposition 4.1 (iii), if $(\mathfrak{B}, \nu)$ is a constructive BA, $\text{ch}(\mathfrak{B}) = (1, 0, 1)$, and the sets $A(\mathfrak{B})$, $\text{Atl}(\mathfrak{B})$, $\text{At}(\mathfrak{B})$ and $I(\mathfrak{B})$ are decidable with respect to ν , then $(\mathfrak{B}, \nu)$ is strongly constructive.

The next theorem is rather unexpected in this context.

Theorem 4.16 [26] *If $(\mathfrak{B}, \nu)$ is a constructive BA, $\text{ch}(\mathfrak{B}) = (1, 0, 1)$, and the sets $\nu^{-1}(A(\mathfrak{B}))$, $\nu^{-1}(\text{Atl}(\mathfrak{B}))$ and $\nu^{-1}(\text{At}(\mathfrak{B}))$ are recursive, then there exists a numeration μ of $\mathfrak{B}$ such that $(\mathfrak{B}, \mu)$ is strongly constructive.*

The proofs of both theorems are based on special algebraic properties of BA's with elementary characteristics $(1, 1, 0)$ and $(1, 0, 1)$. BA's with elementary characteristic $(1, 1, 0)$ admit the following presentation:

Proposition 4.17 *If $\mathfrak{B}$ is a BA with $\text{ch}(\mathfrak{B}) = (1, 1, 0)$, then*

$$\mathfrak{B} \simeq \sum_{i \in \omega} \mathfrak{A}_i \times \mathfrak{B}_i,$$

where $\mathfrak{A}_i$ is atomic, and $\mathfrak{B}_i$ is atomless, $i \in \omega$.

For BA's with characteristic $(1, 0, 1)$, there exists the following isomorphism criterion (Proposition 4.18).

The sequence $A = \{a_0, a_1, \dots\}$ of non-zero atomic elements of a BA $\mathfrak{A}$ is called principal if it satisfies the following conditions.

- (1) $a_i \cap a_j \neq 0$, $i \neq j$
- (2) If $a \in \text{At}(\mathfrak{A})$, then a lies below a finite join of elements of A .
- (3) If $d \in \mathfrak{A}$, then for almost all $i \in \omega$, either $d \cap a_i = 0$, or $a_i \leq d$.

The subset $P_A \subset P(\mathbb{N})$ is then defined in the following way. For $I \subset \mathbb{N}$, let $I \in P_A$ if and only if there exists an element $d \in \mathfrak{A}$ such that $a_i \leq d$ if $i \in I$, and $a_i \cap d = 0$ if $i \notin I$.

Proposition 4.18 *Let $\mathfrak{B}$ and $\mathfrak{D}$ be BA's with $\text{ch}(\mathfrak{B}) = \text{ch}(\mathfrak{D}) = (1, 0, 1)$. $\mathfrak{B}$ and $\mathfrak{D}$ are isomorphic if and only if there exist principal sequences $A = \{a_0, a_1, \dots\}$ of atomic elements of $\mathfrak{B}$ and $C = \{c_0, c_1, \dots\}$ of atomic elements of $\mathfrak{D}$ such that $[a_i]_{\mathfrak{B}} \simeq [c_i]_{\mathfrak{D}}$, $i \in \omega$, and $P_A = P_C$.*

Theorems 4.15 and 4.16 admit a generalization:

Theorem 4.19

- (i) *If $\mathfrak{B}$ is a BA, $\text{ch}(\mathfrak{B}) = (n+1, 1, 0)$, and $\mathfrak{B}$ is $(4n+2)$ -constructivizable, then $\mathfrak{B}$ is strongly constructivizable.*
- (ii) *If $\mathfrak{B}$ is BA, $\text{ch}(\mathfrak{B}) = (n+1, 0, 1)$ and $\mathfrak{B}$ is $(4n+3)$ -constructivizable, then $\mathfrak{B}$ is strongly constructivizable.*

At the end of this section, we formulate several open questions:

- (1) For $n \geq 1$, is there a BA $\mathfrak{B}$ with $\text{ch}(\mathfrak{B}) = (n + 1, 1, 0)$, such that $\mathfrak{B}$ is $(4n + 1)$ -constructivizable, but not strongly constructivizable? ¹
- (2) For $n \geq 1$, is there a BA $\mathfrak{B}$ with $\text{ch}(\mathfrak{B}) = (n + 1, 0, 1)$, such that $\mathfrak{B}$ is $(4n + 2)$ -constructivizable, but not strongly constructivizable?
- (3) when does there exist a BA $\mathfrak{B}$ with $\text{ch}(\mathfrak{B}) = (n_1, n_2, n_3)$, such that $\mathfrak{B}$ is $(4k + 1)$ – $((4k + 2)$ –, $(4k + 3)$ –) constructivizable, but not $(4k + 2)$ – $((4k + 3)$ –, $(4k + 4)$ –) constructivizable?

5 Constructive Boolean algebras with decidability conditions on ultrafilters

The distinctive feature of the results of this section is that they are closely related to the problems of general constructive model theory.

Let $(\mathfrak{M}, \nu)$ be a numerated model of signature σ_0 . Let

$$\sigma_1 \Leftarrow \sigma_0 \cup \{c_i \mid i \in \omega\},$$

where c_i , $i \in \omega$, are new constants, and let $\mathfrak{M}_\nu$ be an enrichment of $\mathfrak{M}$ of signature σ_1 such that the value of c_i in $\mathfrak{M}$ is νi . $F_1(\mathfrak{M}_\nu)$ denotes a Lindenbaum BA of the model $\mathfrak{M}_\nu$, i.e., the algebra of all formulae of signature σ_1 with only one free variable factorized by the equivalence $\sim_{\mathfrak{M}_\nu}$, where

$$\varphi \sim_{\mathfrak{M}_\nu} \psi \Leftarrow \text{Th}(\mathfrak{M}_\nu) \vdash \varphi \leftrightarrow \psi.$$

Let $\gamma : \mathbb{N} \rightarrow F_1(\mathfrak{M}_\nu)$ be a numeration of this BA induced by the Gödel numeration of formulae.

Proposition 5.1 *A numerated model $(\mathfrak{M}, \nu)$ is strongly constructive if and only if the BA $(F_1(\mathfrak{M}_\nu), \gamma)$ is strongly constructive.*

This proposition is complemented by the next fact (see [29]).

Theorem 5.2 *For any strongly constructive atomic BA $(\mathfrak{B}, \nu)$, there exists a strongly constructive model $(\mathfrak{M}, \mu)$ such that $(\mathfrak{B}, \nu)$ is recursively isomorphic to $(F_1(\mathfrak{M}_\mu), \gamma)$.*

¹Added in proof, after the article had been prepared for publication, problem 1 stated in the end of §4 was partially solved by Vlasov and Goncharov (see [40]).

These two results demonstrate the significance of constructive BA's for the constructive model theory.

Ershov [9] has conjectured that any strongly constructive model has a proper strongly constructive effective elementary extension. He has shown that the existence of such an extension of a numerated model $(\mathfrak{M}, \nu)$ is equivalent to the existence of a recursive ultrafilter of the numerated BA $(F_1(\mathfrak{M}_\nu), \gamma)$.

This conjecture has been refuted by Peretyat'kin [29]. His result uses the presentation of BA's via trees.

A set $\Pi \subset 2^{<\omega}$ is called a chain if the relation $\subseteq$ induces a linear order on Π , and $(y \subseteq x \ \& \ x \in \Pi) \implies y \in \Pi$.

Theorem 5.3 *Let $\mathcal{D}$ be an r.e. tree. Then the BA $\mathfrak{C}(\mathcal{D})$ has a constructivization ν such that there exists a one-to-one correspondence between the infinite recursive chains contained in $\mathcal{D}$ and the non-principal recursive ultrafilters of $(\mathfrak{C}(\mathcal{D}), \nu)$.*

There exists a natural correspondence between infinite chains of a tree $\mathcal{D}$ and non-principal ultrafilters of the BA $\mathfrak{C}(\mathcal{D})$:

Let $\varphi : \mathcal{D} \rightarrow \mathfrak{C}(\mathcal{D})$ be an admissible map which determines the BA $\mathfrak{C}(\mathcal{D})$, and Π an infinite chain of $\mathcal{D}$. Then it is easy to show that

$$\mathcal{U}(\Pi) \Leftarrow \{a \in \mathfrak{C}(\mathcal{D}) \mid (\exists \tau \in \mathcal{D})(a \supset \varphi(\tau))\}$$

is a non-principal ultrafilter of $\mathfrak{C}(\mathcal{D})$. If F is a non-principal ultrafilter of $\mathfrak{C}(\mathcal{D})$, then there exists a chain Π of $\mathcal{D}$ such that $F = \mathcal{U}(\Pi)$. Moreover, there is only one chain with this property; it is the unique infinite chain Π such that $\varphi(\Pi) \subset F$. So $\mathcal{U}$ is a one-to-one correspondence. In fact, the theorem asserts that there exists a constructivization ν of $\mathfrak{C}(\mathcal{D})$ such that the map $\mathcal{U}$ preserves the decidability of chains of $\mathcal{D}$ and the decidability of ultrafilters of $\mathfrak{C}(\mathcal{D})$ with respect to ν .

In particular, the constructive BA $(\mathfrak{C}(\mathcal{D}), \nu)$ has a non-principal recursive ultrafilter if and only if $\mathcal{D}$ contains an infinite recursive chain. However, there exists a tree which does not contain any infinite recursive chains [29]. Therefore, we have:

Theorem 5.4 (Peretyat'kin [29]) *There exists a strongly constructive BA $(\mathfrak{B}, \nu)$ that does not contain any non-principal recursive ultrafilters.*

This fact and Theorem 5.2 allow us to obtain the following:

Theorem 5.5 *There exists a strongly constructive model without any proper strongly constructive effective elementary extension.*

There are several generalizations of Theorem 5.4. The tree mentioned above, that does not contain any infinite recursive chain, generates a BA which is isomorphic to the BA of recursive sets $\mathfrak{R}$. In [29], the constructivizations of this BA that contain exactly n decidable non-principal ultrafilters were determined. Note that $\mathfrak{R} \simeq \mathfrak{B}_{\omega \times \eta}$, and $\mathfrak{R}/F(\mathfrak{R})$ is atomless, i.e., $\mathfrak{R}$ is not superatomic. The decidability of ultrafilters in the class of superatomic BA's was studied by Pinus [31].

Theorem 5.6 (Pinus [31]) *Let $\mathfrak{B}$ be a superatomic constructivizable BA, n a natural number, and $\Phi_0, \dots, \Phi_{n-1}$ non-principal ultrafilters of $\mathfrak{B}$. Then there exists a constructivization ν of $\mathfrak{B}$ such that the sets $\nu^{-1}(\Phi_i)$, $i < n$, are recursive, and no other ultrafilter is recursive with respect to ν .*

The following question arises: for which classes of BA's and which classes of constructivizations do the constructions of Theorems 5.4 and 5.6 fail? A partial answer is given by

Theorem 5.7 (Pinus [31]) *If $\mathfrak{B}$ is a strongly constructive BA, then all non-principal ultrafilters of $\mathfrak{B}$ are recursive with respect to any strong constructivization of $\mathfrak{B}$ if and only if the set of all non-principal ultrafilters of $\mathfrak{B}$ is finite, i.e., the factor algebra $\mathfrak{B}/F(\mathfrak{B})$ is finite.*

Goncharov and Nurtazin [20] have examined the constructivity of prime and universal models of complete decidable theories, using the technique from [29]. They focussed their attention to the question of uniform decidability of families of ultrafilters.

The r.e. tree $\mathcal{D}$ constructed in [20] satisfies the following conditions.

- (1) All maximal chains of $\mathcal{D}$ are computable.
- (2) The family of all maximal chains is not computable.
- (3) The family of all finite maximal chains is not computable.

The existence of such a tree, and the relationship between trees and BA's established in Theorem 5.3, allow us to conclude that:

Theorem 5.8 (Goncharov, Nurtazin [20]) *There exists a constructive BA $(\mathfrak{B}, \mu)$ such that*

- (i) *all ultrafilters of $\mathfrak{B}$ are recursive with respect to μ ,*
- (ii) *the family of all ultrafilters of $\mathfrak{B}$ is not computable with respect to μ ,*
- (iii) *the family of all principal ultrafilters of $\mathfrak{B}$ is not computable with respect to μ .*

The following is a stronger version of Theorem 5.2.

Theorem 5.9 *For any constructive BA $(\mathfrak{B}, \nu)$, we can effectively construct a complete decidable theory $T(\mathfrak{B}, \nu)$ in a countable set of unary predicates with equality, such that $F_1(T(\mathfrak{B}, \nu), \gamma)$ is recursively isomorphic to $(\mathfrak{B}, \nu)$.*

It is well known that a complete theory T is totally transcendental if and only if the set of all ultrafilters of $F_1(T)$ is countable. The BA $\mathfrak{B}$ from Theorem 5.8 has only recursive ultrafilters with respect to ν , therefore the set of all ultrafilters of $\mathfrak{B}$ is countable, i.e., $\mathfrak{B}$ is superatomic. Therefore, the corresponding theory $T(\mathfrak{B}, \nu)$ is totally transcendental.

Let $\mathfrak{M}$ be a countable universal model of the theory T . If $\mathfrak{M}$ is strongly constructive, then the family of types that are compatible with T is computable, i.e., the family of all ultrafilters of the BA $F_1(T)$ is computable. The computability of the family of principal ultrafilters of $F_1(T)$ is equivalent to the limit computability of the set of atoms of T . It is shown in [20] that the last assertion is equivalent to the strong constructivizability of the prime model of T . The theories constructed in Theorem 5.9 permit the elimination of quantifiers, because their signature only contains unary predicates. Therefore, we have:

Theorem 5.10 (Goncharov, Nurtazin [20]) *There exists a decidable, totally transcendental theory T such that the prime and universal models of T are not constructivizable.*

The subsequent development of the technique, as used in Theorems 5.8 and 5.10, allows us to prove that the upper bound for the complexity of prime models of complete decidable theories is exact.

Theorem 5.11 (Drobotun [3]) *Let T be a complete decidable theory, and let $\mathfrak{M}$ be a prime model of T . Then $\mathfrak{M}$ is Δ_2^0 -constructivizable.*

This is an upper bound for the complexity. Can we improve this estimate, for example, in terms of m -reducibility? This naturally arising question has a negative answer:

Let $\mathcal{S}$ be a countable family of sets, $\mathcal{S} \subseteq \mathcal{P}(\omega)$. A numeration $\mu : \omega \rightarrow \mathcal{S}$ is called *A-computable* if $\{\langle k, \ell \rangle \mid k \in \mu(\ell)\} \leq_m A$. A family $\mathcal{S}$ is called *A-computable* if there exists at least one A-computable numeration of $\mathcal{S}$.

Note that if the full diagram of $(\mathfrak{M}, \nu)$, where $\mathfrak{M}$ is the prime model of T , is m -reducible to A , then the family of principal 1-types of T is A-computable.

Theorem 5.12 (Drobotun [3]) *For any $A \in \Delta_2^0$, there exists a constructive superatomic BA $(\mathfrak{B}, \nu)$ such that the family of all principal ultrafilters of $(\mathfrak{B}, \nu)$ is not A-computable.*

By analogy with the results mentioned above, we obtain the following:

Corollary 5.13 (Drobotun [3]) *For any set $A \in \Delta_2^0$, there exists a totally transcendental decidable theory T such that $\text{Th}(\mathfrak{M}, \mu) \not\leq_m A$ for any numeration μ of the prime model $\mathfrak{M}$ of T .*

A detailed analysis of the structure of superatomic BA's allows us to establish that the level of complexity Δ_1^1 in the hyperarithmetical Kleene-Mostowski hierarchy is the least upper bound for the complexity of countable saturated models in the class of all decidable totally transcendental theories [19].

Theorem 5.14 (Goncharov, Drobotun [19]) *A saturated model of a decidable totally transcendental theory is Δ_1^1 -strongly constructivizable.*

This is an upper bound, and we have to establish that we cannot improve it.

We have introduced above the notion of a chain in a tree $\mathcal{D}$. Now we define inductively an α -chain (or a chain of the type α) of a tree $\mathcal{D}$, for any countable ordinal α .

- (i) Any finite maximal chain of tree $\mathcal{D}$ has type 0.

- (ii) If for any ordinal $\beta < \alpha$, the notion of β -chain is defined already, then a chain Π has the type α if its type is not already defined and if, for almost all elements $x \in \Pi$, any maximal chain of $\mathcal{D}$ that is different from Π and contains x , has the type $\beta < \alpha$.

The one-to-one correspondence $\mathcal{U}$ between the set of maximal chains of a tree $\mathcal{D}$ and the set of ultrafilters of the BA $(\mathfrak{C}(\mathcal{D}), \nu_{\mathcal{D}})$ has been established in Theorem 5.3. This correspondence has some additional properties.

Proposition 5.15 *A chain Π of a tree $\mathcal{D}$ is an α -chain if and only if $\mathcal{U}(\Pi)$ is an ultrafilter of the type α .*

Proposition 5.16 *If the type of any chain of a tree $\mathcal{D}$ is at most α , and $\mathcal{D}$ has only a finite number of α -chains, then $\mathfrak{C}(\mathcal{D})$ is a superatomic BA of the type (α, k) , where k is the number of distinct α -chains of $\mathcal{D}$.*

The exactness of the upper estimate mentioned above follows from:

Proposition 5.17 *For any $C \in \Delta_1^1$, there exists an r.e. tree $\mathcal{D}$ such that $C \leq_T \Pi$, where Π is a chain of maximal type in $\mathcal{D}$.*

Corollary 5.18 *For any $C \in \Delta_1^1$, there exists a constructive superatomic BA $(\mathfrak{B}, \nu)$ such that $C \leq_T \nu^{-1}(\mathcal{F})$, where $\mathcal{F}$ is an ultrafilter of maximal type in $\mathfrak{B}$.*

As a conclusion, we have:

Theorem 5.19 *For any set $C \in \Delta_1^1$, there exists a decidable totally transcendental theory T such that $C \leq_T \gamma^{-1}(\text{Th}(\mathfrak{M}, \nu))$ for any numeration ν of a countable saturated model $\mathfrak{M}$ of T .*

We conclude this section with a result about the constructivity of the Morley rank α_T [2] of a Δ_1^1 -decidable totally transcendental theory T .

The following is a relativization of a theorem from [14].

Theorem 5.20 *If $\mathfrak{B}$ is a $\Delta_1^{1,A}$ -constructivizable superatomic BA, then the ordinal type of $\mathfrak{B}$ is A -constructive.*

Corollary 5.21 *A superatomic BA $\mathfrak{B}$ is $\Delta_1^{1,A}$ -constructivizable if and only if $\mathfrak{B}$ is A -constructivizable.*

This result shows that, contrary to the general case (see [27]), the hyperarithmetical hierarchy of superatomic BA's is degenerative.

It is well known that the BA $F_1(T)$ of a totally transcendental theory T is superatomic. Moreover, the Morley rank α_T of T is equal to $\beta + 1$, where β is the ordinal type of $F_1(T)$. Therefore, we have:

Theorem 5.22 *If T is a complete $\Delta_1^{1,A}$ -decidable totally transcendental theory, then its Morley rank is an A -constructive ordinal.*

This theorem generalizes the result of Sacks about the relative constructivity of the Morley rank [36].

6 Constructivity beyond first order logic

In this short section, we collect several independent results about the $\mathcal{F}$ -constructivizability of BA's, where $\mathcal{F}$ is some fragment of the second order language. Although progress in this direction was not systematic, there are a few interesting theorems which should be mentioned. All the results stated here are due to Pinus [30, 31].

Let us consider the extension $L(Q)$ of the first order language by a quantifier Q meaning "there exist infinitely many elements such that ...". This formalism, introduced first by Mostowski [24], can be placed between the first order language and the weak second order language. It is quite obvious that $L(Q)$ is not equivalent to the first order language. For example, the $L(Q)$ -theory of the BA $\mathfrak{B}_\omega$ [23] is ω -categorical while, at the same time, the BA $\mathfrak{B}_\omega$ is elementary equivalent to any infinite atomic BA because it has elementary characteristic $(0, \omega, 0)$.

The next result about the strong constructivizability of ω -atomic BA's is well known.

Theorem 6.1 (Goncharov [15]) *Any ω -atomic BA $\mathfrak{B}$ is strongly constructivizable if and only if $\mathfrak{B}$ is constructivizable.*

This theorem has an even stronger version:

Theorem 6.2 (Pinus [30]) *Any constructivizable ω -atomic BA is $L(Q)$ -constructivizable.*

The proof is based on the fact that, on the class of all BA's, any $L(Q)$ -formula is equivalent to a suitable first order formula of signature

$$\sigma^* \Leftarrow \langle \cup, \cap, c, 0, 1, \Phi_1(x), \Phi_2(x), \Phi_3(x) \rangle,$$

where $\Phi_1(x)$ means that x is an atomic element containing infinitely many atoms, $\Phi_2(x)$ means that x is a join of a finite number of atoms and atomless elements, and $\Phi_3(x)$ means that x is a join of a finite number of atoms, i.e., an element of the Fréchet ideal. Thus, the quantifier Q can be eliminated, and we can deduce the following fact:

Theorem 6.3 (Pinus [30]) *The $L(Q)$ -theory of the class of all BA's is decidable.*

The theory of BA's of signature σ^* is interpreted in the weak second order theory of the class of linearly ordered sets, and this class is decidable, as established by Läuchli [23]. Therefore the first theory is also decidable.

Let us consider yet another extension of first order language. Let $\mathcal{I}$ denote the restriction of the second order language of the signature of BA's such that $\mathcal{I}$ admits only unary predicate variables which are interpreted on BA's only as ideals of BA's. The details can be found in [32]. We will mention two facts about $\mathcal{I}$ -constructive BA's:

Theorem 6.4 (Ershov [9]) *Any constructivizable superatomic BA is $\mathcal{I}$ -constructivizable.*

Theorem 6.5 (Ershov [9]) *A countable atomless BA is $\mathcal{I}$ -constructivizable.*

This concludes our study of the constructivity of BA's in various extensions of first order language.

References

- [1] J. Barwise, (ed.), Handbook of Mathematical Logic, Stud. Logic Found. Math., **90** (1977).
- [2] C. C. Chang and H. J. Keisler, Model Theory, 3rd. edn., Stud. Logic Found. Math., **73** (1990); [1st. edn. 1973, 2nd. edn. 1977].

- [3] B. N. Drobutun, Enumerations of simple models (Russian), *Sibirsk. Math. Zh.*, **18** (1977) 1002–1014, 1205; [translated in: *Siberian Math. J.*, **18** (1977) 707–716].
- [4] V. D. Dzgoev, On the constructivizability of Boolean algebras (Russian), in: *Proc. 4th. All-Union Conf. Math. Logic*, Kishinev, (1976).
- [5] V. D. Dzgoev, Constructive Boolean algebras (Russian), *Mat. Zametki*, **44** (1988) 750–757, 861; [translated in: *Math. Notes*, **44** (1988) 896–901].
- [6] A. Ehrenfeucht, Applications of games to the completeness problem for formalized theories, *Fund. Math.*, **49** (1960/61) 129–141.
- [7] Yu. L. Ershov, Decidability of the elementary theory of relatively complemented distributive lattices, and of the theory of filters (Russian), *Algebra i Logika*, **3** (1964) 17–38.
- [8] Yu. L. Ershov, A hierarchy of sets I (Russian), *Algebra i Logika*, **7** (1968) 47–74 [translated in: *Algebra and Logic*, **7** (1968) 25–43].
- [9] Yu. L. Ershov, Constructive models (Russian), in: *Izbr. Vopr. Alg. i Log.*, [Selected Questions in Algebra and Logic], *Sbornik posvjascen. pamjati A. I. Mal'ceva*, [A collection dedicated to the memory of A. I. Mal'tsev], *Izdat. Nauk Sibirsk. Otdel.*, Novosibirsk, (1973) 111–130.
- [10] Yu. L. Ershov, Theory of Numerations (Russian), *Monographs in Math. Logic and Foundations of Math.*, (Nauka, Moskva, 1977).
- [11] Yu. L. Ershov, Decision Problems and Constructivizable Models (Russian), (*Mathematical Logic and Foundations of Mathematics*, Nauka, Moskva, 1980).
- [12] L. J. Feiner, Hierarchies of Boolean algebras, *J. Symbolic Logic*, **35** (1970) 365–374.
- [13] R. Fraïssé, Sur quelques classifications des systèmes de relations, *Publ. Sci. Univ. Alger. Sér. A.*, **1** (1954) 35–182.
- [14] S. S. Goncharov, The constructivizability of superatomic Boolean algebras (Russian), *Algebra i Logika*, **12** (1973) 31–40, 120; [translated in: *Algebra and Logic*, **12** (1973) 17–22].

- [15] S. S. Goncharov, Certain properties of the constructivization of Boolean algebras (Russian), *Sibirsk. Math. Zh.*, **16** (1975) 264–278; [translated in: *Siberian Math. J.*, **16** (1975) 203–214].
- [16] S. S. Goncharov, Restricted theories of constructive Boolean algebras (Russian), *Sibirsk. Math. Zh.*, **17** (1976) 797–812; [translated in: *Siberian Math. J.*, **17** (1976) 601–611].
- [17] S. S. Goncharov, Countable Boolean Algebras (Russian), (Nauka. Sibirsk. Otdel., Novosibirsk, 1988).
- [18] S. S. Goncharov and B. N. Drobotun, The generalization of constructivizations (Russian), 14th. Vsesojuz. Algebra. Konf., Novosibirsk, **2** (1977) 93–94.
- [19] S. S. Goncharov and B. N. Drobotun, Numerations of saturated and homogeneous models (Russian), *Sibirsk. Math. Zh.*, **21** (1980) 25–41, 236; [translated in: *Siberian Math. J.*, **21** (1980) 164–176].
- [20] S. S. Goncharov and A. T. Nurtazin, Constructive models of complete decidable theories (Russian), *Algebra i Logika*, **12** (1973) 125–142, 243; [translated in: *Algebra and Logic*, **12** (1973) 67–77].
- [21] V. Harnik and M. Makkai, Applications of Vaught sentences and the covering theorem, *J. Symbolic Logic*, **41** (1976) 171–187.
- [22] A. H. Lachlan, On the lattice of recursively enumerable sets, *Trans. Amer. Math. Soc.*, **130** (1968) 1–37.
- [23] H. Läuchli, A decision procedure for the weak second order theory of linear order, in: *Contributions to Mathematical Logic*, (Colloq., Hanover, Aug. 1966), H. A. Schmidt, K. Schütte and H.-J. Thiele, (eds.), *Stud. Logic Found. Math.*, (1968) 189–197.
- [24] A. Mostowski, On a generalization of quantifiers, *Fund. Math.*, **44** (1957) 12–36.
- [25] A. T. Nurtazin, Strong and weak constructivizations and computable families (Russian), *Algebra i Logika*, **13** (1974) 311–323, 364; [translated in: *Algebra and Logic*, **13** (1974) 177–184].
- [26] S. P. Odintsov, Restricted theories of constructive Boolean algebras of the lower stratum (Russian), *Inst. of Math., Novosibirsk*, (Preprint No. 21) (1986).

- [27] S. P. Odintsov and V. L. Selivanov, The arithmetical hierarchy and ideals of enumerated Boolean algebras (Russian), *Sibirsk. Math. Zh.*, **30** (1989) 140–149; [translated in: *Siberian Math. J.*, **30** (1989) 952–960].
- [28] E. A. Paljutin, Boolean algebras with a categorical theory in a weak second order logic (Russian), *Algebra i Logika*, **10** (1971) 523–534; [translated in: *Algebra and Logic*, **10** (1971) 325–331 (1973)].
- [29] M. G. Peretyat'kin, Strongly constructive models and enumerations of the Boolean algebra of recursive sets (Russian), *Algebra i Logika*, **10** (1971) 535–557; [translated in: *Algebra and Logic*, **10** (1971) 332–345].
- [30] A. G. Pinus, Theory of Boolean algebras in a calculus with the quantifier “there exists infinitely many” (Russian), *Sibirsk. Math. Zh.*, **17** (1976) 1417–1421, 1440.
- [31] A. G. Pinus, Constructivization of Boolean algebras (Russian), *Sibirsk. Math. Zh.*, **22** (1981) 169–175, 231; [translated in: *Siberian Math. J.*, **22** (1981) 616–620 (1982)].
- [32] M. O. Rabin, Decidability of second-order theories and automata on infinite trees, *Bull. Amer. Math. Soc. (N.S.)*, **74** (1968) 1025–1029.
- [33] J. B. Remmel, Recursive Boolean algebras, in: *Handbook of Boolean Algebras*, Vol. 3, J. D. Monk, R. Bonnet and S. Koppelberg, (eds.), (North-Holland, Amsterdam, NY, 1989), Ch. 25, 1097–1165.
- [34] L. J. Richter, Degrees of structures, *J. Symbolic Logic*, **46** (1981) 723–731.
- [35] H. Rogers Jr., *Theory of Recursive Functions and Effective Computability*, (1st. edn., McGraw-Hill, New York-Toronto, Ont.-London, 1967; 2nd. edn., MIT Press, Cambridge, Mass., London, 1987).
- [36] G. E. Sacks, *Saturated Model Theory*, (Mathematics Lecture Note Series, W. A. Benjamin, Inc., Reading, Mass., 1972).
- [37] V. L. Selivanov, Algorithmic complexity of algebraic systems (Russian), *Mat. Zametki*, **44** (1988) 823–832, 863; [translated in: *Math. Notes*, **44** (1988) 944–950].
- [38] A. D. Taimanov, The characteristics of axiomatizable classes of models (Russian), *Algebra i Logika Sem.*, **1** (1962) 5–31.

- [39] A. Tarski, Arithmetical classes and types of Boolean algebras, Preliminary Report, *Bull. Amer. Math. Soc. (N.S.)*, **55** (1949) 64.
- [40] V. N. Vlasov and S. S. Goncharov, Strong constructibility of Boolean algebras with elementary characteristic $(1,1,0)$ (Russian), *Algebra i Logika*, **32** (1993) 618–630; [translated in: *Algebra and Logic*, **32** (1993) 334–341].

S. P. Odintsov
Institute of Mathematics,
Universitetski pr.4,
Novosibirsk, 630090
Russia

Chapter 21

Reverse Algebra

S. G. Simpson* and J. Rao

Contents

Introduction

1. Subsystems of second order arithmetic and their models
2. Reverse algebra
3. Relationship with recursive algebra

References

Introduction

Reverse algebra is part of a program called *reverse mathematics* whose goal is to answer the following Main Question: “Which set existence axioms are needed to prove the theorems of ordinary mathematics?”. By ordinary mathematics we mean non-set-theoretic mathematics, i.e., those parts of mathematics which do not essentially depend on the abstract theory of uncountable ordinal and cardinal numbers. Thus, ordinary mathematics includes: geometry, number theory, calculus, differential equations, real and complex analysis, countable algebra, countable combinatorics, the topology of complete separable metric spaces, and the theory of separable Banach spaces. On the other hand, ordinary mathematics does not include abstract functional analysis, abstract set theory, general topology or universal algebra. Therefore, the goal of reverse algebra is to answer the following Main Question: “Which set existence axioms are needed in order to prove the theorems of countable algebra?”. The set existence axioms which we consider are formulated in the context of weak subsystems of the second-order arithmetic.

Almost all countable algebra can be developed within the formal system $\mathbf{Z}_2$ of second-order arithmetic (Hilbert-Bernays [16]). Subsequent investigations revealed that the set existence axioms of $\mathbf{Z}_2$ are in fact much too strong.

*Simpson’s research was partially supported by NSF grant DMS-9303478.

It turns out that weak subsystems of $\mathbf{Z}_2$, which usually employ much weaker set existence axioms, are sufficient for the development of the bulk of countable algebra. In the course of the systematic study of weak subsystems of $\mathbf{Z}_2$ and the investigation of the Main Question, a very interesting phenomenon happens very often: if a theorem of countable algebra is proved from the weakest possible set existence axioms, it will be possible to “reverse” the algebraic theorem by proving that it is equivalent to those axioms over a weaker base theory. We refer to this phenomenon as *Reverse Algebra*.

1 Subsystems of second order arithmetic and their models

The language of second order arithmetic, denoted L_2 , is a two-sorted first order language with number variables $i, j, m, n, \dots$, and set variables $X, Y, Z, \dots$. The number variables are intended to range over the set of natural numbers $\mathbb{N} = \omega$, while the set variables are intended to range over subsets of ω . Numerical terms are built up as usual from number variables, constant symbols 0 and 1, and binary operation symbols $\cdot$ and $+$. Atomic formulas are $t_1 = t_2$, $t_1 < t_2$, and $t_1 \in X$, where t_1 and t_2 are numerical terms. Formulas are built up from atomic formulas by means of propositional connectives, number quantifiers $\forall n$ and $\exists n$, and set quantifiers $\forall X$ and $\exists X$.

The language of first order arithmetic, denoted L_1 , is the same as L_2 with the omission of the set variables and set quantifiers and the $\in$ symbol. Thus L_1 is just the familiar language of first order Peano arithmetic, denoted PA.

If t is any numerical term and φ is any formula, we write $(\forall m < t)\varphi$ (respectively $(\exists m < t)\varphi$) as an abbreviation for $\forall m(m < t \rightarrow \varphi)$ (respectively $\exists m(m < t \wedge \varphi)$). The expressions $(\forall m < t)$ and $(\exists m < t)$ are known as bounded numerical quantifiers. A formula is Σ_0^0 if all of its quantifiers are bounded numerical quantifiers. A formula is Σ_1^0 (respectively Π_1^0) if it is of the form $\exists m \theta$ (respectively $\forall m \theta$), where θ is Σ_0^0 . A formula is arithmetical if it does not contain any set quantifier. A formula is Π_1^1 if it is of the form $\forall X \theta$, where θ is arithmetical.

For $k = 0, 1$, Σ_k^0 -Induction is the scheme

$$(\varphi(0) \wedge \forall n(\varphi(n) \rightarrow \varphi(n+1))) \longrightarrow \forall n \varphi(n)$$

where φ is Σ_k^0 . Also Σ_k^0 -Comprehension is the scheme

$$\exists X \forall n (n \in X \leftrightarrow \varphi(n))$$

where φ is Σ_k^0 and X does not occur in φ . Finally, Δ_k^0 -Comprehension is the scheme

$$\forall n (\varphi(n) \leftrightarrow \varphi(n)) \longrightarrow \exists X \forall n (n \in X \leftrightarrow \varphi(n))$$

where φ is Σ_k^0 , φ is Π_k^0 , and X does not occur in φ . Δ_1^0 -Comprehension is also called Recursive Comprehension.

An L_2 -structure is an ordered 7-tuple

$$M = (|M|, S_M, +_M, \cdot_M, 0_M, 1_M, <_M),$$

where $|M|$ is a set which serves as the range of the number variables, and S_M is a collection of subsets of $|M|$ which serves as the range of the set variables. The first order part of M is the L_1 -structure

$$(|M|, +_M, \cdot_M, 0_M, 1_M, <_M).$$

An ω -model is an L_2 -structure whose first order part is standard; that is, $(\omega, +, \cdot, 0, 1, <)$. Thus, ω -models can be identified with collections of subsets of ω ; that is, the range of set variables. If T is any L_2 -theory, the first order part of T is the L_1 -theory whose theorems are exactly the those L_1 -formulas which are theorems of T .

The formal system of second order arithmetic, denoted $\mathbf{Z}_2$ (or $\Pi_\infty^1 - \text{CA}_0$), consists of the ordered semi-ring axioms for $(\omega, +, \cdot, 0, 1, <)$, together with the Induction axiom

$$(0 \in X) \wedge \forall n ((n \in X) \rightarrow (n + 1 \in X)) \longrightarrow \forall n (n \in X)$$

and the Comprehension scheme

$$\exists X \forall n (n \in X \leftrightarrow \varphi(n))$$

where φ is any L_2 -formula in which X does not occur freely. The intended model of $\mathbf{Z}_2$ is $(\omega, P(\omega), +, \cdot, 0, 1, <)$, where $P(\omega)$ is the set of all the subsets of ω .

Most of the known subsystems of $\mathbf{Z}_2$ are of little or no interest from the narrow point of view of our investigation of the Main Question. With only a little exaggeration, we may say that there are only six systems which are relevant to countable algebra. These six systems are the following:

The system: RCA_0 .

The letters RCA stand for Recursive Comprehension Axioms. The system RCA_0 consists of the ordered semi-ring axioms for $(\omega, +, \cdot, 0, 1, <)$, together with Δ_1^0 -Comprehension and Σ_1^0 -Induction. Roughly speaking, the set existence axioms of RCA_0 are only strong enough to prove the existence of recursive sets of natural numbers. From the viewpoint of countable algebra, the best way to view RCA_0 is a kind of formalized recursive algebra. In terms of the Main Question, RCA_0 serves as a weak base theory, i.e., a vantage point from which we can appreciate the non-recursive content of other parts of algebra. The minimum ω -model of RCA_0 consists of all recursive subsets of ω . In general, an ω -model of RCA_0 is any nonempty collection of subsets of ω which is closed under join and relative recursiveness.

The first order part of RCA_0 is the L_1 -theory Σ_1 -Induction, which is just first-order Peano Arithmetic with the induction scheme restricted to Σ_1 -formulas. Also, it is known that RCA_0 is a conservative extension of Primitive Recursive Arithmetic (denoted PRA) with respect to Π_2 -sentences; that is, for every Π_2 -sentence σ , $\text{RCA}_0 \vdash \sigma$ if and only if $\text{PRA} \vdash \sigma$.

Within RCA_0 , we can prove that the class of all total functions is closed under composition, primitive recursion and μ -operator. So the exponential function can be defined in RCA_0 , and we can encode finite sequences of natural numbers by natural numbers. The set of codes for finite sequences of 0 and 1, denoted $2^{<\mathbb{N}}$, is definable inside RCA_0 . If n is a natural number and $f : \mathbb{N} \rightarrow \{0, 1\}$ a total function, by $f[n]$ we mean the finite sequence $\langle f(0), f(1), \dots, f(n-1) \rangle \in 2^{<\mathbb{N}}$. For $s, t \in 2^{<\mathbb{N}}$, we write $s \subseteq t$ to mean that s is an initial segment of t . A tree is a set $T \subseteq 2^{<\mathbb{N}}$ such that $\forall s \forall t (s \subseteq t \wedge t \in T) \rightarrow s \in T$. A path through T is a total function $f : \mathbb{N} \rightarrow \{0, 1\}$ such that $\forall n (f[n] \in T)$.

The system: WKL_0 .

The letters WKL stand for Weak König's Lemma, which is the statement that every infinite tree of $2^{<\mathbb{N}}$ has an infinite path. The system WKL_0 consists of the axioms of RCA_0 together with Weak König's Lemma. The ω -models of WKL_0 are Scott sets. If $(|M|, +_M, \cdot_M, 0_M, 1_M, <_M)$ is a nonstandard

model of PA, and let $SSy(M)$ consist of those subsets X of ω such that $X = \{n \in \omega : M \models \theta(n, \bar{a})\}$ where $\theta(x, \bar{y})$ is an L_1 -formula and $\bar{a} \in |M|$, then $SSy(M)$ is a Scott set (see Kaye [18]) and

$$(\omega, SSy(M), +_\omega, \cdot_\omega, 0_\omega, 1_\omega, <_\omega)$$

is an ω -model of WKL_0 . By the Kreisel Basis Theorem (see Shoenfield [20]), there exist nonstandard models M of PA such that M is Δ_2^0 . Therefore, there exist Δ_2^0 ω -models of WKL_0 . There is no minimal ω -model of WKL_0 . The minimum ω -model of RCA_0 is not a model of WKL_0 , but is the intersection of all the ω -models of WKL_0 . Hence WKL_0 is properly stronger than RCA_0 .

The first-order part of WKL_0 is also the L_1 -theory Σ_1 -Induction. In fact, WKL_0 is a conservative extension of RCA_0 with respect to Π_1^1 -sentences (Harrington, see Simpson [10]). Hence the logic strength of WKL_0 is the same as RCA_0 ; that is, they have the same proof-theoretic ordinal ω^ω . Therefore, WKL_0 is also conservative over PRA with respect to Π_2^0 -sentences.

The system: ACA_0 .

The letters ACA stand for Arithmetical Comprehension Axioms. The system ACA_0 consists of those axioms of RCA_0 together with the scheme of Arithmetical Comprehension $\exists X \forall n (n \in X \leftrightarrow \varphi(n))$ where φ is any arithmetical formula in which X does not occur freely. If

$$M = (|M|, S_M, +_M, \cdot_M, 0_M, 1_M, <_M)$$

is a model of ACA_0 , then the first-order part of M is a model of PA. On the other hand, if $(|M|, +_M, \cdot_M, 0_M, 1_M, <_M)$ is a model of PA, and let S_M consist of those subsets X of $|M|$ such that

$$X = \{a \in |M| : (|M|, +_M, \cdot_M, 0_M, 1_M, <_M) \models \varphi(a)\},$$

for some arithmetic formula $\varphi(y)$ with parameters from $|M|$ and no free variables other than y , then $M = (|M|, S_M, +_M, \cdot_M, 0_M, 1_M, <_M)$ is a model of ACA_0 . Therefore, the first order part of ACA_0 is first order Peano Arithmetic. The minimum ω -model of ACA_0 consists of just the arithmetical subsets of ω . In general, an ω -model of ACA_0 is any nonempty collection of subsets of ω which is closed under Turing reducibility, join, and Turing jump.

König's Lemma, which is the statement that every infinite, finitely branching tree of $\mathbb{N}^{<\mathbb{N}}$ has an infinite path, is provable in ACA_0 (Friedman [15]). Thus, ACA_0 contains WKL_0 . On the other hand, if

$$(|M|, +_M, \cdot_M, 0_M, 1_M, <_M)$$

is a Δ_2^0 nonstandard model of PA, then $(\omega, SSy(M), +, \cdot, 0, 1, <)$ is a model of WKL_0 but is not a model of ACA_0 . Therefore ACA_0 is properly stronger than WKL_0 .

The system: ATR_0 .

The letters ATR stand for Arithmetical Transfinite Recursion, the axiom which says that arithmetical comprehension can be iterated along any countable well ordering. The system ATR_0 consists of the axioms of ACA_0 together with the Arithmetical Transfinite Recursion axiom. Every ω -model of ATR_0 includes HYP , the set of all hyperarithmetical subsets of ω . Since HYP properly includes the minimum ω -model of ACA_0 , it follows that the latter is not an ω -model of ATR_0 . Therefore, ATR_0 is properly stronger than ACA_0 .

The system: $\Pi_1^1 - CA_0$.

This is the system of Π_1^1 -Comprehension; that is, it consists of the axioms of RCA_0 together with the scheme of Π_1^1 -Comprehension, which is the comprehension scheme restricted to Π_1^1 -formulas. $\Pi_1^1 - CA_0$ is properly stronger than ATR_0 ; one way to see this is to note that the complete Π_1^1 subset of ω belongs to any β -model of $\Pi_1^1 - CA_0$, but there exist β -models of ATR_0 which do not contain this set. (A β -model is defined to be an ω -model which is a Σ_1^1 -elementary submodel of the full model, $P(\omega)$. Note that any β -model is automatically a model of ATR_0 .)

The system: RCA_0^* .

The weakest system we are going to consider in this survey is RCA_0^* . The language of RCA_0^* is the language of second-order arithmetic L_2 augmented by a binary function symbol $\exp$ denoting exponentiation. The system RCA_0^* consists of the ordered semi-ring axioms for $(\omega, +, \cdot, 0, 1, <)$, two axioms for $\exp$: $\exp(m, 0) = 1$; $\exp(m, n + 1) = \exp(m, n) \cdot m$, the scheme of Σ_0^0 -Induction and the scheme of Δ_1^0 -Comprehension. The numerical terms of this extended language are those of L_2 together with $t_1^{t_2} (= \exp(t_1, t_2))$. Atomic formulas and formulas are built as before. Trivially, RCA_0 is equivalent to RCA_0^* plus Σ_1^0 -Induction.

RCA_0^* is properly weaker than RCA_0 . The following is a model of RCA_0^* which is not a model of RCA_0 (Simpson-Smith[11]): Let

$$(|M|, +_M, \cdot_M, 0_M, 1_M, <_M)$$

be any nonstandard model of the first-order Peano Arithmetic and $a \in |M|$ be any nonstandard integer. Define a sequence $b_0 = a$, $b_{n+1} = b_n^{b_n}$, and let $|I|$ be the set of all $b \in |M|$ such that $b < b_n$ for some $n \in \omega$. Clearly $|I|$ is an initial segment of M which is closed under $+$, $\cdot$ and $\exp$. If we let S_I be the set of all the subsets of $|I|$ of the form $X \cap |I|$ where X is encoded in M by a single element (i.e., $X \subset |M|$ and $M \models \exists n \forall i (i \in X \leftrightarrow p_i \text{ divides } n)$, where p_i is the i -th prime number), then

$$I = (|I|, S_I, +_I, \cdot_I, 0_I, 1_I, <_I, =_I)$$

becomes a model of RCA_0^* . Note that the total function defined by primitive recursion as $f(0) = a$; $f(n+1) = f(n)^{f(n)}$ is not closed in I (because $f(a) \notin I$), I is not a model of RCA_0 ; that is, Σ_1^0 -Induction fails.

The first-order part of RCA_0^* is $B\Sigma_1 + \text{EXP}$ (Simpson-Smith [11]). Along with RCA_0 , RCA_0^* also serves as a weak base theory in reverse mathematics.

2 Reverse algebra

We now are going to sketch, by two examples, how some theorems of countable algebra can be developed in weak subsystems of $\mathbf{Z}_2$, and how to “reverse” them.

We begin with the numerical pairing function

$$(m, n) = \frac{(m+n)(m+n+1)}{2} + n.$$

If X and Y are subsets of $\mathbb{N} = \{n : n = n\}$, we define

$$X \times Y = \{(m, n) : m \in X \wedge n \in Y\}.$$

Thus, $\mathbb{N} \times \mathbb{N} \subset \mathbb{N}$.

In order to define the set $\mathbb{Z}$ of *integers*, we first define an equivalence $=_{\mathbb{Z}}$ on $\mathbb{N} \times \mathbb{N}$ by putting $(m, n) =_{\mathbb{Z}} (p, q)$ if and only if $m + q = n + p$. We then put $\mathbb{Z} = \{(m, n) : (m, n) \text{ is minimal in its } =_{\mathbb{Z}}\text{-class}\}$. We define $+$, $-$, $\cdot$, $\dots$ on $\mathbb{Z}$ so that

$$\begin{aligned} (m, n) + (p, q) &=_{\mathbb{Z}} (m + p, n + q), \\ -(m, n) &=_{\mathbb{Z}} (n, m), \\ (m, n) \cdot (p, q) &=_{\mathbb{Z}} (mp + nq, mq + np), \quad \text{etc..} \end{aligned}$$

One then can prove in RCA_0^* that the system $(\mathbb{Z}, +, -, \cdot, 0, 1, <)$ has the usual properties of an ordered integral domain, the Euclidean property, etc.. In order to define the set $\mathbb{Q}$ of *rational numbers*, we let $\mathbb{Z}^+ = \{a \in \mathbb{Z} : a > 0\}$, and define an equivalence relation $=_{\mathbb{Q}}$ on $\mathbb{Z} \times \mathbb{Z}^+$ by putting $(a, b) =_{\mathbb{Q}} (c, d)$ if and only if $a \cdot_{\mathbb{Z}} d = b \cdot_{\mathbb{Z}} c$. We then put

$$\mathbb{Q} = \{(a, b) \in \mathbb{Z} \times \mathbb{Z}^+ : (a, b) \text{ is minimal in its } =_{\mathbb{Q}}\text{-class}\}.$$

We define $+, -, \cdot, \dots$ on $\mathbb{Q}$ so that

$$\begin{aligned} (a, b) + (c, d) &=_{\mathbb{Q}} (ad + bc, bd), \\ -(a, b) &=_{\mathbb{Q}} (-a, b), \\ (a, b) \cdot (c, d) &=_{\mathbb{Q}} (ac, bd), \text{ etc..} \end{aligned}$$

One can prove in RCA_0^* that the system $(\mathbb{Q}, +, -, \cdot, 0, 1, <)$ has the usual properties of an ordered field, etc..

A countable field F consists of a subset $|F| \subset \mathbb{N}$, together with binary operations $+_F$ and $\cdot_F$, and a unary operation $-_F$, and distinguished elements 0_F and 1_F , such that the system $(|F|, +_F, -_F, \cdot_F, 0_F, 1_F)$ obeys the usual field axioms, e.g., $\forall x \forall y (x \cdot y = y \cdot x)$ and $\forall x (x \neq 0 \rightarrow \exists y (x \cdot y = 1))$, etc.. Countable rings and countable vector spaces can be defined in similar way. We now are ready to have our first example.

Theorem 2.1 (Friedman) Σ_1^0 -Induction (therefore RCA_0) is equivalent to the algebraic theorem: Every finitely generated vector space over a countable field has a basis, the equivalence being provable in RCA_0^* .

Proof. We work in RCA_0^* .

Firstly we assume Σ_1^0 -Induction. Let V be a finitely generated vector space over a countable field F , and b_0 a (code of) finite generating set of V . The set of all (codes of) finite generating sets of V is Σ_1^0 -definable using b_0 as a parameter: b is a (code of) a finite generating set of V if and only if

$$(\forall i < \text{length}(b_0)) (\exists c) \left((b_0)_i = \sum_{k=0}^{\text{length}(b)} (c)_j (b)_j \right).$$

So, by Σ_1^0 -least element principle, which is equivalent to Σ_1^0 -Induction over RCA_0^* (Simpson-Smith [11]), let n be the least possible size that a generating

set might have. It is easy to see that any generating set of size n is a basis of V .

Conversely, we are going to prove Bounded Σ_1^0 -Comprehension: given any Σ_1^0 -formula $\varphi(x)$ and a natural number n , the set $\{k < n : \varphi(k)\}$ exists, which is equivalent to Σ_1^0 -Induction over RCA_0^* (see [12]). Let $\theta(x)$ be the Σ_1^0 -formula $(x < n) \wedge \varphi(x)$. By Lemma 2.4 of [12], there exists a set of natural numbers X and a one-to-one function $f : X \rightarrow \mathbb{N}$ such that

$$\forall k (\theta(k) \leftrightarrow \exists j \in X (k = f(j))).$$

Let V_0 be the set of formal sums $\sum_{i=0}^{2n-1} q_i x_i$, $q_i \in \mathbb{Q}$, where $\mathbb{Q}$ is the set of all rational numbers. Clearly V_0 is a finite dimensional vector space over $\mathbb{Q}$ and the set $\{x_i : i < 2n\}$ is a basis of it. For each $k \in X$, put $x'_k = x_{2f(k)} + kx_{2f(k)+1}$, and let V be the subspace of V_0 generated by $X' = \{x'_k : k \in X\}$. V exists by Δ_1^0 -Comprehension, since $v = \sum_{i=0}^{2n-1} q_i x'_i \in V$ if and only if

$$(\forall j < n) ((q_{2j} \neq 0 \rightarrow f(q_{2j+1}/q_{2j}) = j).$$

Moreover, X' is a basis of V . We now can form the quotient space V_0/V , the elements of V_0/V are the least coset representatives, here “least” refers to the ordering of $\mathbb{N}$. V_0/V is a finitely generated vector space over $\mathbb{Q}$, and by (2), it has a basis X'' . Clearly $\bar{X} = X' \cup X''$ is a basis of V_0 . Now it is not hard to see that for any natural number k , $\theta(k)$ if and only if at least one of the unique expressions of x_{2k} and x_{2k+1} in terms of the basis $\bar{X}$ involves x'_j such that $k = f(j)$, which can be formalized by a Π_1^0 -formula. So, the set $\{k < n : \varphi(k)\}$ exists by Δ_1^0 -Comprehension. $\square$

An immediate corollary of Theorem 2.1 is that RCA_0^* is not strong enough to prove the algebraic theorem: Every finitely generated vector space over a countable field has a basis. Note that RCA_0 is not strong enough to prove the following algebraic theorem: every countable vector space has a basis. The later is equivalent to ACA_0 , the equivalence being provable in RCA_0^* (see [1, 12]).

Now let R be a countable commutative ring. An ideal of R is a subset $I \subset R$ such that $0_R \in I$ and $1_R \notin I$, and

$$\forall a \forall b (a \in I \wedge b \in I \rightarrow a +_R b \in I), \quad \text{and}$$

$$\forall r \forall a (r \in R \wedge a \in I \rightarrow r \cdot_R a \in I).$$

A maximal ideal of R is an ideal I such that

$$\forall r (r \in R \wedge r \notin I \rightarrow \exists s (s \in R \wedge r \cdot_R s -_R 1_R \in I)).$$

The following is our second example.

Theorem 2.2 (Friedman-Simpson-Smith) *ACA₀ is equivalent to the algebraic theorem: Every countable commutative ring has a maximal ideal, the equivalence being provable in RCA₀.*

Proof. We work in RCA₀.

First we assume ACA₀. Let R be a commutative ring. For any $X \subset R$, say X is *good* if X does not generate R as an R -module; that is, 1 is not of the form $\sum_{i=1}^n s_i a_i$ where $s_i \in R$, $a_i \in X$. Let $\{r_n : n \in \mathbb{N}\}$ be an enumeration of the elements of R . Define $f : \mathbb{N} \rightarrow \{0, 1\}$ by $f(n) = 0$ if $\{r_i : i < n \wedge f(i) = 0\} \cup \{r_n\}$ is *good*, $f(n) = 1$ otherwise. Existence of f follows by Arithmetical Comprehension. Let $M = \{r_m : f(m) = 0\}$, then I is a maximal ideal of R .

Conversely, we want to show that the range of every one-to-one $f : \mathbb{N} \rightarrow \mathbb{N}$ exists, which is equivalent to ACA₀ (see [1]). Given $f : \mathbb{N} \rightarrow \mathbb{N}$, let $R_0 = \mathbb{Q}[\{x_n : n \in \mathbb{N}\}]$ be the polynomial ring with countably many indeterminates, and F_0 the field of fractions of R_0 . Let $\varphi(b)$ be the Σ_1^0 formula asserting that $b = r/s$, $r \in R_0$ and s is of the form $qx_{f(m_1)}^{e_1} \cdots x_{f(m_k)}^{e_k}$ with $q \in \mathbb{Q}$, $q \neq 0$. By Lemma 2.4 of [10], there exists a countable commutative ring R and a one-to-one function $h : R \rightarrow K_0$ such that $\forall b (\varphi(b) \leftrightarrow \exists a (h(a) = b))$. By the assumption, let M be a maximal ideal of R . It is not hard to verify that $\forall n (\exists m (f(m) = n) \leftrightarrow h^{-1}(x_n) \notin M)$. By Δ_1^0 -Comprehension, the set $X = \{n : h^{-1}(x_n) \notin M\}$ exists. Clearly X is the range of f . $\square$

The following is a list of results of reverse algebra.

Theorem 2.3 (Friedman-Simpson-Smith [1]) *The following algebraic theorems are provable in RCA₀:*

- (1) *Every countable abelian group has a divisible closure.*
- (2) *Every countable field has an algebraic closure.*
- (3) *Every countable ordered field has a unique real closure.*

- (4) *For each countable field F , and $f(x) \in F[x]$, there exists a splitting field for $f(x)$ over F .*
- (5) (Primitive Element Theorem) *Every finite separable extension of a countable field is a simple extension.*

Theorem 2.4 *Within RCA_0^* , one can prove that the following algebraic theorems are equivalent to Σ_1^0 -Induction, therefore equivalent to RCA_0 :*

- (1) (Friedman; see Hatzikiriakou [3]) *Every finitely generated vector space over a countable field has a basis.*
- (2) (Hatzikiriakou [3]) *Every torsion-free, finitely generated abelian group is free.*
- (3) (Hatzikiriakou [3]) *The Fundamental Structure Theorem for Finitely Generated Abelian Groups.*
- (4) (Simpson-Smith [12]) *For each countable field F and every $f(x) \in F[x]$, $f(x)$ has only finitely many roots in F .*
- (5) (Simpson-Smith [12]) *For each countable field F and every $f(x) \in F[x]$, $f(x)$ has an irreducible factor.*
- (6) (Simpson-Smith [12]) *For each countable field F , $F[x]$ is a unique factorization domain.*
- (7) (Rao-Simpson [9]) *For each countable field F , $F[x]$ is a principal ideal domain.*
- (8) (Rao-Simpson [9]) *Every countable Euclidean domain is a unique factorization domain.*
- (9) (Rao-Simpson [9]) *Every countable Euclidean domain is a principal ideal domain.*
- (10) Gauss's Theorem (Rao-Simpson [9]) *If R is a countable unique factorization domain, so is the polynomial ring $R[x]$.*
- (11) (Friedman-Simpson-Smith [1], Rao [8]) *The Fundamental Theorem of Galois Theory.*

Theorem 2.5 *Within RCA_0 , one can prove that WKL_0 is equivalent to each of the following algebraic theorems.*

- (1) (Friedman-Simpson-Smith [1]) *Every countable field has a unique algebraic closure.*
- (2) Artin-Schreier Theorem (Friedman-Simpson-Smith [1]) *Every countable formally real field is orderable.*
- (3) (Friedman-Simpson-Smith [1]) *Every countable commutative ring has a prime ideal.*
- (4) Levi's Theorem (Hatzikiriakou-Simpson [7]) *A countable abelian group is orderable if and only if it is torsion-free.*
- (5) The Extension Theorem for Valuations (Hatzikiriakou-Simpson [6]) *Given a monomorphism of countable fields $h : F \rightarrow K$ and a valuation ring V of F , there exists a valuation ring U of K such that $h^{-1}(U) = V$.*

Theorem 2.6 *Within RCA_0 , one can prove that ACA_0 is equivalent to each of the following algebraic theorems.*

- (1) (Friedman-Simpson-Smith [1]) *Every countable vector space over $\mathbb{Q}$ (or over any countable field) has a basis.*
- (2) (Friedman-Simpson-Smith [1]) *Every countable commutative ring has a maximal ideal.*
- (3) (Hatzikiriakou [4]) *Every countable commutative ring has a minimal prime ideal.*
- (4) (Friedman-Simpson-Smith [1]) *Every countable abelian group has a unique divisible closure.*
- (5) (Friedman-Simpson-Smith [1]) *Every countable divisible abelian group is injective.*
- (6) (Friedman-Simpson-Smith [1]) *Every countable abelian group has a subgroup consisting of all the torsion elements.*

- (7) (Friedman-Simpson-Smith [1]) *Every countable field has a transcendence base.*
- (8) (Friedman-Simpson-Smith [1]; see Simpson [11]) *Every countable field is isomorphic to a subfield of a countable algebraically closed field.*
- (9) (Friedman-Simpson-Smith [1]; see Simpson [11]) *Every countable ordered field is isomorphic to a subfield of a countable real closed field.*

Theorem 2.7 (Friedman-Simpson-Smith [1]) *Within RCA_0 , one can prove that ATR_0 is equivalent to Ulm's Theorem: any two countable reduced abelian p -groups which have the same Ulm invariants are isomorphic.*

Theorem 2.8 (Friedman-Simpson-Smith [1]) *Within RCA_0 , one can prove that Π_1^1 -Comprehension is equivalent to the algebraic theorem: every countable abelian group is a direct sum of a divisible group and a reduced group.*

Over some weak base theories, such as RCA_0 , we also can prove that some combinatorial principles are equivalent to some algebraic theorems. The following is a list of results of this kind.

Theorem 2.9 (Simpson [10], Hatzikiriakou [2]) *Within RCA_0 , one can prove that the following theorems are equivalent.*

- (1) Hilbert Basis Theorem *For each countable field F and each $n \in \mathbb{N}$, the polynomial ring $F[x_1, \dots, x_n]$ is Hilbertian.*
- (2) *For each countable field F and each $n \in \mathbb{N}$, the ring of formal power series $F[[x_1, \dots, x_n]]$ is Hilbertian.*
- (3) *The ordinal number ω^ω is well ordered.*

Theorem 2.10 (Simpson [10]) *Within RCA_0 , one can prove that the following theorems are equivalent.*

- (1) Robson Basis Theorem *For each countable field F and each $n \in \mathbb{N}$, the non-commutative ring $F\langle x_1, \dots, x_n \rangle$ is Robsonian.*
- (2) *The ordinal number ω^{ω^ω} is well ordered.*

3 Relationship with Recursive Algebra

We now turn to a discussion of how reverse algebra is related to recursive algebra. For general background on recursive algebra, the reader may consult Rabin [10], Fröhlich-Shepherdson [15], Metakides-Nerode [19], and other references which are given in this volume.

First of all, our goals in reverse algebra are quite different from the goals of recursive algebra. As explained in previous sections, our investigations are guided by a very definite purpose, that is, to find out what set existence axioms are needed to prove the theorems of countable algebra. Recursive algebraists are concerned with the truth or falsity of certain recursive analogs of well known algebraic theorems. We, on the other hand, are concentrating on provability or non-provability of the theorems themselves, in the presence of certain set existence axioms. Clearly the goals of the recursive algebraists are quite different from ours. Nevertheless, some of the details of our work are parallel to recursive algebra.

The similarities and differences between reverse algebra and recursive algebra may be illustrated by an example. Consider the well-known Artin-Schreier theorem, which says that every formally real field is orderable. Ershov [12] constructs a recursive formally real field which is not recursively orderable. He views this construction as providing a “recursive counterexample”, which shows that the Artin-Schreier theorem is “recursively false”. What this really means, is that a certain statement about recursive algebraic structures, although analogous to the Artin-Schreier theorem, is false. Our treatment of the Artin-Schreier theorem is quite different. We show (Theorem 2.5(2)) that the Artin-Schreier theorem for countable fields is provably equivalent to WKL_0 over RCA_0 . An immediate corollary is that the set existence axioms of WKL_0 suffice to prove the Artin-Schreier theorem for countable fields, while those of RCA_0 alone do not suffice.

Having said this, we cheerfully acknowledge that Ershov’s idea is a key ingredient in our proof that the Artin-Schreier theorem for countable fields implies WKL_0 . Ershov encodes a disjoint, recursively inseparable pair of recursively enumerable sets into a recursive formally real field in such way that any ordering of the field would separate the pair. We observe that a relativized form of Ershov’s construction can be carried out provably in RCA_0 . This observation is an essential part of our proof. Thus we owe a considerable debt to recursive algebra.

We may attempt to make a payment on our debt, by pointing out that some of our results can be partially reformulated in terms of *degrees of unsolvability*. The point is that a recursive algebraist, having discovered that a certain ordinary algebraic theorem is “recursively false”, may go on to wonder about the degree of unsolvability of the object whose existence is “known classically”. Information of this type follows readily from our results.

Generally, let $\theta(U, V)$ be an arithmetical formula, and suppose that the Π_2^1 -sentence $\sigma \equiv \forall U \exists V \theta(U, V)$ is known from our investigations to be provably equivalent to WKL_0 over RCA_0 . It follows immediately that σ is false in the minimum ω -model of RCA_0 . This means that there exists a recursive set U such that $\theta(U, V)$ is false for all recursive sets V . In other words, we have a “recursive counterexample” to σ . But more can be said. Let $\mathbf{d}$ be any degree of unsolvability such that $\mathbf{d} \gg \mathbf{0}$ (see Simpson [22] for a brief summary of known results about such degrees). Then we can say that for all recursive U , there exists V such that $\theta(U, V)$, and V is recursive in $\mathbf{d}$. In other words, given that the “problem” U is recursive, we have tight upper bounds on the extent to which the “solution” V is nonrecursive. These upper bounds have been investigated thoroughly by recursion theorists. Thus, it follows from a result of Jockusch and Soare [17] that we can find a V which is *low*, that is, the halting problem relative to V as an oracle is recursive relative to the ordinary halting problem as an oracle.

For an example illustrating the above general discussion, let σ be the Π_2^1 -sentence which is the statement of the Artin-Schreier theorem. Theorem 2.5(2) not only implies the existence of a “recursive counterexample”, but also implies that when a formally real field is recursive, it has an ordering whose degree of unsolvability is low. This additional information cannot be obtained by the methods of recursive algebra alone.

The other half of the Theorem 2.5(2), which says that the Artin-Schreier theorem is provable in WKL_0 , has no counterpart in recursive algebra. At the same time, there are certain other aspects of recursive algebra which are not reflected in our investigation. For example, Metakides-Nerode [19] show that any Π_1^0 subclass of 2^ω is recursively homeomorphic to the space of orderings of some recursive field. This refinement of Ershov’s construction [13] has no counterpart in our work.

In general, every algebraic theorem provable in RCA_0 is true in the minimum ω -model of RCA_0 , and is therefore “recursively true”. Any algebraic theorem whose proof needs set existence axioms stronger than RCA_0 , such as WKL_0 , ACA_0 , ATR_0 or $\Pi_1^1 - \text{CA}_0$, will be false in the minimum ω -model of

RCA_0 , and this provides a recursive counterexample. Thus subsystems of the second-order arithmetic provide a finer classification of well known algebraic theorems. From the standpoint of recursive algebra, such a theorem τ may be classified as either “recursively true” or “recursively false”, meaning that the recursive analog of τ is true or false, that is, that τ is true or false in the minimum ω -model of RCA_0 . From the standpoint of subsystems of $\mathbf{Z}_2$, τ can be more finely classified according to which set existence axioms are needed to prove it. In particular, if τ is “recursively false”, hence not provable in RCA_0 , we may still ask whether it is provable in some stronger systems such as WKL_0 , ACA_0 , ATR_0 or $\Pi_1^1 - \text{CA}_0$.

Even for those theorems which are provable in RCA_0 (and therefore “recursively true”), we may still ask whether they are provable in RCA_0^* or not. For example, RCA_0^* is strong enough to prove that the polynomial ring $F[x]$ is a Euclidean domain for every countable field (Rao-Simpson [9]), but is not strong enough to prove that $F[x]$ is a unique factorization domain for every countable field. To prove the latter, Σ_1^0 -Induction is needed, therefore RCA_0 is needed (Theorem 2.4 (6)).

We hope that these remarks have helped to clarify the the relationship between recursive algebra on the one hand, and reverse algebra on the other hand.

References

We first list the publications which touch on reverse algebra, [1]–[12]. This part of the bibliography is intended to be fairly complete. We then list the other works which have been cited above.

Friedman-Simpson-Smith [1] is the original work on reverse algebra. Simpson [24] is a very good survey. For a detailed exposition of subsystems of $\mathbf{Z}_2$ and reverse mathematics, including proofs of the most of the results which were merely stated above, see Simpson’s forthcoming book [11].

Works on Reverse Algebra:

- [1] H. M. Friedman, S. G. Simpson and R. L. Smith, Countable algebra and set existence axioms, *Ann. Pure Appl. Logic*, **25** (1983) 141–181; addendum, *ibid.* **27** (1985) 319–320.
- [2] K. Hatzikiriakou, Commutative algebra in subsystems of second order arithmetic, Ph.D. Thesis, Penn. State Univ., University Park, PA, (1989).

- [3] K. Hatzikiriakou, Algebraic disguises of Σ_1^0 induction, *Arch. Math. Logic*, **29** (1989) 47–51.
- [4] K. Hatzikiriakou, Minimal prime ideals and arithmetic comprehension, *J. Symbolic Logic*, **56** (1991) 67–70.
- [5] K. Hatzikiriakou, WKL_0 and Stone’s separation theorem for convex sets, *Ann. Pure Appl. Logic*, **77** (1996) 245–249.
- [6] K. Hatzikiriakou and S. G. Simpson, Countable valued fields in weak subsystems of second-order arithmetic, *Ann. Pure Appl. Logic*, **41** (1989) 27–32.
- [7] K. Hatzikiriakou and S. G. Simpson, WKL_0 and orderings of countable abelian groups, in: *Logic and Computation*, (Proc. Workshop, CMU, Pittsburgh, June 30 – July 2, 1987), W. Sieg, (ed.), *Contemp. Math.*, **106** (1990) 177–180.
- [8] J. Rao, Some topics in reverse algebra, Ph.D. Thesis, Penn. State Univ., University Park, PA, (in preparation).
- [9] J. Rao and S. G. Simpson, Factorization in subsystems of second order arithmetic, (in preparation).
- [10] S. G. Simpson, Ordinal numbers and the Hilbert basis theorem, *J. Symbolic Logic*, **53** (1988) 961–974.
- [11] S. G. Simpson, Subsystems of Second Order Arithmetic, (in preparation).
- [12] S. G. Simpson and R. L. Smith, Factorization of polynomials and Σ_1^0 Induction, *Ann. Pure Appl. Logic*, **31** (1986) 289–306.

Other Cited Works:

- [13] Yu. L. Ershov, *Theorie der Numerierungen*, III, (translated from Russian and edited by G. Asser and H.-D. Hecker), *Z. Math. Logik Grundlag. Math.*, **23** (1977) 289–371.
- [14] H. M. Friedman, Higher set theory and mathematical practice, *Ann. Math. Logic*, **2** (1971) 325–357.
- [15] A. Fröhlich and J. C. Shepherdson, Effective procedures in field theory, *Philos. Trans. Roy. Soc. London, Ser. A*, **248** (1956) 407–432.

- [16] D. Hilbert and P. Bernays, *Grundlagen der Mathematik*, Vol. I, II, Springer, Berlin, (1934, 1939); [2nd. edn. (1968, 1970)].
- [17] C. G. Jockusch Jr. and R. I. Soare, Π_1^0 classes and degrees of theories, *Trans. Amer. Math. Soc.*, **173** (1972) 33–56.
- [18] R. Kaye, *Models of Peano Arithmetic*, Oxford Logic Guides, **15**, (Oxford Univ. Press, NY, 1991).
- [19] G. Metakides and A. Nerode, Effective content of field theory, *Ann. Math. Logic*, **17** (1979) 289–320.
- [20] M. O. Rabin, Computable algebra, general theory and theory of computable fields, *Trans. Amer. Math. Soc.*, **95** (1960) 341–360.
- [21] J. R. Shoenfield, *Mathematical Logic*, (Addison-Wesley, 1967).
- [22] S. G. Simpson, Degrees of unsolvability: a survey of results, in: J. Barwise, (ed.), *Handbook of Mathematical Logic*, Stud. Logic Found. Math., **90** (1977) 631–652.
- [23] S. G. Simpson, Which set existence axioms are needed to prove the Cauchy/Peano theorem for ordinary differential equations?, *J. Symbolic Logic*, **49** (1984) 783–802.
- [24] S. G. Simpson, Subsystems of $\mathbb{Z}_2$ and reverse mathematics, appendix to *Proof Theory*, 2nd. edn., (by G. Takeuti) (North Holland, Amsterdam, 1987), 432–446.

Stephen G. Simpson
Department of Mathematics
Pennsylvania State University
University Park, PA 16802
e-mail: simpson@math.psu.edu

Ju Rao
Department of Mathematics
Pennsylvania State University
University Park, PA 16802
e-mail: simpson@math.psu.edu